

Teppo Toriseva

ELINTARVIKEOHJELMAN SEURANTAJÄRJESTELMÄ

Liiketoiminta Huittinen
Tietojenkäsittelyn koulutusohjelma
2010



ELINTARVIKEOHJELMAN SEURANTAJÄRJESTELMÄ

Toriseva, Teppo
Satakunnan ammattikorkeakoulu
Tietojenkäsittelyn koulutusohjelma
Marraskuu 2010
Ohjaaja: Kuisma, Pekka
Sivumäärä: 62
Liitteitä: 1

Asiasanat: verkko-ohjelmointi, Flash, tietokonegrafiikka, elintarvikeohjelma, seurantajärjestelmät

Tämän opinnäytetyön aiheena oli Satakunnan elintarvikeohjelman seurantajärjestelmän suunnittelu ja toteutus. Tavoitteena oli suunnitella ja toteuttaa järjestelmä, jolla saadaan kerättyä erilaisia tietoja Satakunnan elintarviketuottajilta, joiden tulokset saadaan näytettyä suoraan graafisina kaavioina. Lopuksi tarkasteltiin järjestelmän toimintaa asiakkaan näkökulmasta.

Seurantajärjestelmä on Internetympäristössä julkaistava sovellus, johon kerätään eri toimialoilla työskenteleviltä yrityksiltä tietoja erilaisista koulutus- ja hankintamenoista. Järjestelmän hienoin ominaisuus on välitön tietojen käsittely heti tallennuksen jälkeen. Tallennuksen jälkeen kaikki yritykset voivat katsoa kaavioita, jotka on luotu käyttäen tietokannassa olevia tietoja ja Flash-sovellusta.

Opinnäytetyössä suunniteltiin ja toteutettiin Internet-verkossa toimiva seurantajärjestelmä Satakunnan elintarvikeohjelman käyttöön. PHP:n, MySQL:n ja Flash-grafiikan avulla toteutettiin ohjelma, jonka avulla yritysten on mahdollista syöttää tietoja suoraan tietokantaan ja joka tuottaa yritykselle vertailutietoina alan keskiarvoja graafisina kuvina. Jokainen tietoja syöttävä yritys näkee seurantajärjestelmästä vain omat vuosittain syöttämänsä luvut ja toimialansa keskiarvon omiin lukuihin verrattuna.

MONITORING SYSTEM FOR FOOD PROGRAM

Toriseva, Teppo
Satakunta University of Applied Sciences
Data processing training program
November 2010
Supervisor: Kuisma, Pekka
Number of pages: 62
Appendices: 1

Keywords: web-programming, Flash, computer graphics, food programme, follow-up systems

The subject of this thesis was to plan and create a monitoring system for a food programme in Satakunta province. My priorities were to create a system to collect different kinds of information from a number of food producers in the Satakunta region, and a programme to create Flash charts based on the information collected from the food producers. Finally, the system was observed from the client's viewpoint.

The monitoring system is a programme working in the WWW-environment, which collects information of educational and procurement expenditures from a number of companies working in different sectors. The system's most advanced feature is instant data processing after the information has been stored. After storing the information, the companies can access the Flash charts, which are produced with data delivered from the database and the Flash programme.

As a result of this thesis, a programme was created by using the PHP programming language. The programme provides companies with an opportunity to deliver their data directly to the database and to produce graphical charts with a comparison to their sector's average values.

The objective of this thesis was to plan and create a programme that works in the Internet and suits the needs of the Satakunta food programme. Companies providing information can see only their own annual values with average values from the sector in comparison to their own values. The programme was created by using the PHP programming language that uses the MySQL database and Flash graphics as a base.

SISÄLLYS

1	JOHDANTO	8
2	OPINNÄYTETYÖSUUNNITELMA.....	8
2.1	Kohteena olevan ongelman kuvaus	9
2.2	Opinnäytetyön taustaa	9
2.3	Opinnäytetyöongelma	9
2.4	Viitekehys	10
2.5	Käytettävät menetelmät ja aineiston kerääminen	11
2.5.1	Käytettävät menetelmät	12
2.5.2	Aineiston kerääminen	12
2.6	Aikataulu	12
3	TIETOJÄRJESTELMÄN KEHITTÄMINEN.....	13
3.1	Evoluutiomalli	13
3.2	PHP-ohjelmointikieli	14
3.3	MySQL-tietokanta	18
3.4	JSON-formaatti.....	20
3.5	Graafisia esityksiä Internetissä - Flash-kaaviot.....	25
3.5.1	JSON-formaatti tiedonlähde Flash-kaaviolle	25
3.5.2	Esimerkki Flash-kaaviosta	33
3.5.3	Esimerkki järjestelmän Flash-kaaviosta.....	36
4	SEURANTAJÄRJESTELMÄ.....	42
4.1	Määrittely	42
4.2	Suunnittelu	43
4.3	Toteutus.....	44
4.4	Toteutustapa	44
4.4.1	Tietokanta	45
4.4.2	Etusivu.....	51
4.4.3	Syöttölomake	52
4.4.4	Grafiikkasivu	52
5	TULOKSET.....	53
5.1	Järjestelmän käyttö.....	53
5.2	Järjestelmän etusivu	53

5.3	Järjestelmän syöttölomake.....	54
5.4	Järjestelmän Omat tiedot-sivu.....	57
5.5	Järjestelmän kaaviosivusto	58
6	JOHTOPÄÄTÖKSET JA YHTEENVETO	60
7	TULEVAISUUDEN NÄKYMIÄ.....	61
8	LOPPUTULOKSEN ARVIOINTI.....	61
	LÄHDELUETTELO	62
	LIITTEET	

SYMBOLI- JA TERMILUETTELO

GNU GPL	Vuonna 1989 luotu vapaa ohjelmistolisenssi ja sen tunnusmerkkejä ovat: vapaus käyttää ohjelmistoa mihin haluaa, vapaus muokata ohjelmistoa miten haluaa, vapaus levittää ohjelmistoa ilmaisena tai maksullisena, vapaus levittää ohjelmiston muokattuja versioita
PHP	Rekursiivinen lyhenne sanoista PHP: Hypertext Preprocessor. Dynaamisten kotisivujen teossa käytetty proseduraalinen skriptimäinen ohjelmointikieli.
MySQL	Suosittu SQL-tietokannan hallintajärjestelmä
PHP/MySQL-ympäristö	Palvelinohjelmistokokonaisuus dynaamisten ja tietokannallisten kotisivujen julkaisuun.
Flash	Adobe Systemsin tuottama kehitysympäristö
JSON	JavaScript Object Notation on yksinkertainen tiedonsiirtomuoto
ODBC	(Open Database Connectivity) Microsoftin määrittelemä rajapinta tietokannoille.
Flash-kaavio	Kaavio, joka on toteutettu käyttäen Flash-ympäristöä ActionScript-ohjelmointikoodilla, johon tiedot on syötetty käyttäen JavaScriptiä.
tag	ohjelmointikoodin aloitus- ja lopetusmerkeistä käytetty nimitys
ActionScript 3	ECMAScriptiin (standardoituun JavaScriptiin) perustuva skriptauskieli.
URL	Uniform Resource Locator (yksilöllisen tietolähteen paikan-taja)
JavaScript	Web-ympäristössä käytettävä komentosarjakieli.

IDE	integrated development environment
HTML	Hypertext Markup Language, hypertekstin merkinäkieli
CSS	Cascading Style Sheets, porrastetut tyyliarkit
md5	message-digest-algoritmi, jota käytetään muun muassa kryptografiassa, esim. Internetissä salasanojen suojaukseen.
hash	Tiiviste eli hajautusarvo, joka saadaan tiivistämällä tieto pienempään tilaan, että sitä voidaan verrata myöhemmin uudestaan aikaisemman saman tiedon oikeellisuuteen.

1 JOHDANTO

Tämä työ sai alkunsa Pyhäjärvi-instituutin Satakunnan elintarvikeohjelma 2007-2013 -projektista, jossa tarvittiin seurantajärjestelmää satakuntalaisten elintarvikeyritysten kehityksen seurantaan.

SAMK oli mukana projektissa ja eräässä projektipalaverissa tuli esiin ajatus suunnitella Internetin kautta päivitettävä seurantaohjelmisto, jota kautta yritykset voisivat päivittää elintarvikealan tietojaan. Toteutunut tietokanta palvelisi näin koottuna tietovarastona Satakunnan elintarvikealaa. Projektissa mukana SAMK:n puolelta oli tohtori Ari Pitkäranta, lehtori Leena Sääski ja lehtori Pekka Kuisma.

Näin syntyi ajatus luoda järjestelmä, jonka avulla voitaisiin Internetin kautta syöttää tietoja ja luoda koostegrafiikkaa kerätyistä tiedoista.

Työn ideointivaiheessa pohdimme, mikä olisi paras työkalu ongelman ratkaisemiseen. Tämän kehittälyhaasteen otin vastaan ja näin tämä opinnäytetyö sai alkunsa.

2 OPINNÄYTETYÖSUUNNITELMA

Opinnäytetyön ongelmana on suunnitella ja toteuttaa Satakunnan elintarvikeohjelman seurantajärjestelmä, jolla seurataan eri toimialayritysten kuluja vuosittain. Järjestelmä luodaan Internet-ympäristöön.

Opinnäytetyön empiirisessä osiossa pyrin kertomaan Satakunnan elintarvikejärjestelmän toteutuksesta prosessina. Työssä käydään läpi koko toteutusprosessi aina määrittelystä, suunnittelun kautta järjestelmän toteutukseen ja valmiiseen tuotteeseen.

Työn teoriaosassa tarkastellaan järjestelmän suunnitteluun liittyviä asioita, esitellään PHP/MySQL-ympäristö sekä Internet-ympäristö ja tutustutaan Flash-grafiikan käyttöön.

Työn lopputuloksena syntyy Internet-ympäristöön ohjelma, joka olisi mahdollista ottaa käyttöön Pyhäjärvi-instituutin Satakunnan elintarvikeohjelma 2007-2013 -projektiin tai johonkin muuhun yritykseen vastaavanlaisten tietojen analysointiin.

2.1 Kohteena olevan ongelman kuvaus

Opinnäytetyöni aiheena on luoda Internetissä toimiva seurantajärjestelmä Satakunnan elintarvikeohjelmalle, joka laadittiin osaamiskeskusrahoituksella maakunnan elintarvikealan toimijoiden yhteistyönä Pyhäjärvi-instituutin koordinoimana vuosille 2007-2013. Opinnäytetyö pitää sisällään seurantajärjestelmän suunnittelun, toteutuksen ja testauksen. Järjestelmän tiedot tallennetaan tietokantaan.

2.2 Opinnäytetyön taustaa

Pyhäjärvi-instituutin projektissa luotiin elintarvikeyrityksille valmis tietojen syöttämistä varten tehty lomake, jolla on tarkoitus kerätä elintarvikeyrityksiltä tietoja jokaisesta kehityskustannuksesta. Ajatuksena on, että kukin yritys syöttää tiedot itse. Tässä opinnäytetyössä toteutus lähti tämän lomakkeen pohjalta. Sen ympärille olisi suunniteltava palvelu, joka tuottaisi koostegrafiikoita syötetyistä tiedoista vertailutietoina.

2.3 Opinnäytetyöongelma

Tämän opinnäytetyön tavoitteena on suunnitella ja toteuttaa Internet-verkossa toimiva ja käytettävyydeltään hyvä seurantajärjestelmä Satakunnan elintarvikeohjelman käyttöön. Jokainen tietojen syöttävä yritys näkee seurantajärjestelmästä vain omat vuosittain syöttämänsä luvut ja toimialansa keskiarvon verrattuna omiin lukuihin. Ohjelma tehdään PHP-ohjelmointikielellä hyödyntäen pohjalla MySQL-tietokantaa ja Flash-grafiikkaa.

Seurantajärjestelmän tiedoista saadaan SQL-kyselyiden avulla erilaisia graafisia esityksiä, joissa on esitetty asiat keskiarvoina. Yritys voi verrata omia lukujaan suhteessa toimialan keskimääräiseen arvoon prosesseittain, mutta kyseinen yritys ei näe toisten yritysten yksittäisiä tietoja. Tämä järjestelmä rakennetaan työn empiriaosassa.

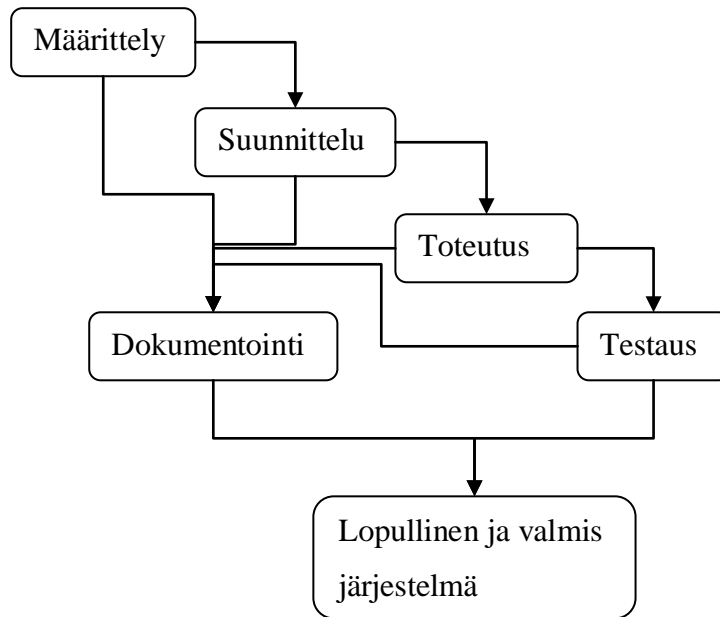
Työn teoriaosassa perehdytään tietojärjestelmän suunnitteluun, PHP-ohjelmointikieleen ja MySQL-tietokannan ideaan sekä Internet-ympäristöön (PHP/MySQL-palvelinympäristöön) että dynaamisen grafiikan luontiin.

Opinnäytetyölläni pyrin vastaamaan siihen seurantatietojen keräämisen ja jalostamisen tarpeeseen, joka Satakunnan elintarvikeohjelmalla on olemassa. Syntynyttä ohjelmaa ja sen ideaa voidaan jatkossa jalostaa vastaavan tyyppisten ongelmien ratkaisussa.

Työn ongelmia ovat tietojen keräys ja syöttö, tietojen käsittely ja miten saada nämä tiedot esitettyä graafisina esityksinä.

2.4 Viitekehys

Tämän opinnäytetyön viitekehysten muodostaa ohjelmiston tietotekninen suunnittelu ja sovelluksen rakentaminen sekä siihen liittyvä tietojen keräys ja jalostaminen graafisiksi esityksiksi. Opinnäytetyöni tarkoituksena on saada elintarvikeyritysten antamat tiedot järjestelmään ja sieltä graafisina esityksinä muiden katseltavaksi.



Kuvio 1. Tietojärjestelmän määrittelyn prosessi

Kuviossa 1 on esitetty opinnäytetyön määrittelyn prosessi.

2.5 Käytettävät menetelmät ja aineiston kerääminen

Koska sovellus tulee suuren ja hajautetun asiakaskunnan käyttöön, on helpompaa tehdä sovellus Internet-ympäristöön keskitetysti. Näin ollen on soveliaampaa käyttää ohjelmoinnissa dynaamista PHP-ohjelmointikieltä tietojen tallennukseen palvelimelle ja tietojen siirtämiseen Flash-sovellukselle.

Flash-grafiikkaa varten tarvitsin sovelluksen, joka osaisi muuttaa tietokannasta haetun tiedon graafiseksi esitykseksi. Kyseisiä sovelluksia löytyy sekä maksullisina että avoimella koodilla varustettuna. Avoinen koodin sovellus sopii työhöni paremmin kuin maksullinen. Eräs ystäväni antoi vinkin yhdestä mahdollisesta avoimen koodin sovelluksesta: <http://teethgrinder.co.uk/open-flash-chart/>. Ihastuin sovelluksen yksinkertaiseen tapaan esittää tieto graafisena esityksenä. Ja sovellus tukee useita eri ohjelmointikieliä.

2.5.1 Käytettävät menetelmät

Tämän opinnäytetyön menetelmiä ovat tietojärjestelmän suunnittelu, PHP-ohjelmointikielen ja siihen liittyvän Flash-grafiikan käyttö. Lähdekirjallisuutena käytetään tietoteknisen suunnittelun osalta kirjallisuutta sekä PHP-kielen käytön osalta Internet-sivustoja. Flash-grafiikkasovelluksen dokumentaatio on melkein aina sovelluskohmainen, koska nämä sovellukset eivät seuraa mitään ennalta määrättyjä määritteitä. Joku voi tehdä sovelluksen joka tukee pelkästään ASP-ohjelmointia, kun taas toinen voi tehdä kaikille ohjelmointikielille yhteensopivan. Ongelma voi olla myös tiedonsyötössä, toiset käyttävät JSON-formaatissa syötettävää tietoa ja toiset ihan jotain omaa tekstiformaattia.

2.5.2 Aineiston kerääminen

Taustamateriaalina työssä käytän aiheeseen liittyvää kirjallisuutta sekä Internet-lähteitä. PHP- ja MySQL -sovellusten kanssa tarvitaan sovellusten omia käsikirjoja.

Flash-kaavioiden teossa käytetään pääasiallisesti Internet-lähteitä. Ongelma Flash-kaavioissa on siinä, että jokainen sovellus on aina erilainen. Yksikään ei ole täysin yhteensopiva toisen toteutustavan kanssa. Tämän vuoksi käytän lähteenä ainoastaan Flash-sovelluksen tekijän omia sivuja.

2.6 Aikataulu

Tämän opinnäytetyön valmistumisaikataulu viivästyi vuodella, koska pidin väli vuoden ja olin sen työelämässä.

3 TIETOJÄRJESTELMÄN KEHITTÄMINEN

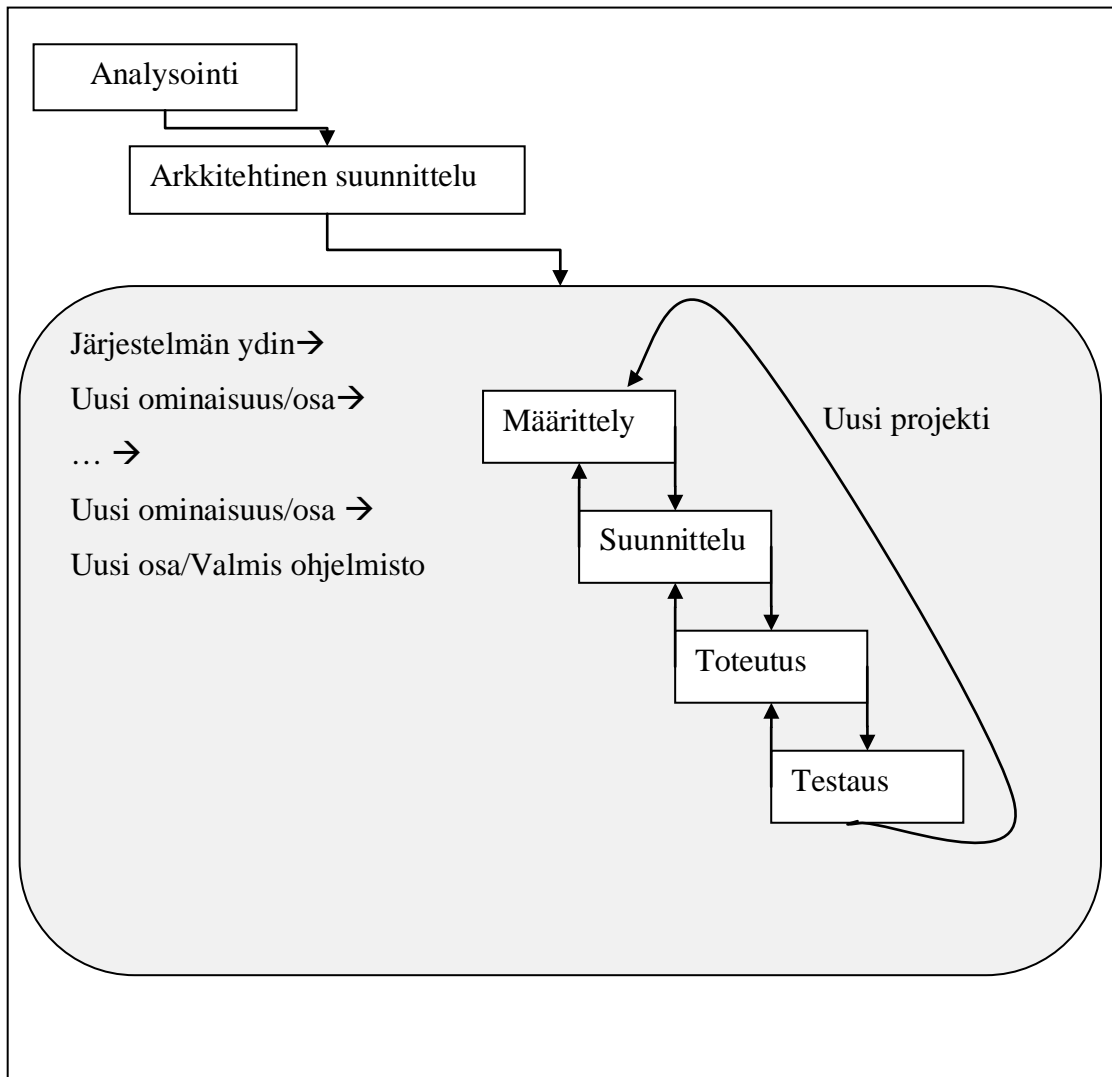
Seuraavissa kappaleissa esittelen erilaisia tietoteknisen suunnittelun malleja ja valitsen niistä sopivan tai jonkin sopivan yhdistelmän, jota voin käyttää tämän järjestelmän suunnitteluun.

3.1 Evoluutiomalli

Evoluutiomalli on toinen käytössä oleva vaihejakomalli vesiputousmallin jälkeen. Joissain tapauksissa evoluutiomallia on sanottu myös inkrementaaliseksi malliksi. Evoluutiomallissa lopputulos rakennetaan osa osalta. Yksittäiset projektit ovat tyypillisesti lyhyitä, muutamasta viikosta muutamaa kuukauteen. Jokaisen projektin jälkeen saadaan tuloksena uusilla ominaisuuksilla tai osilla kasvatettu järjestelmä. Näin tehtynä, ohjelmiston käyttökelpoisuutta päästään testaamaan jo projektin alkuvaiheessa ja mahdolliset ongelmat pystytään vielä ratkomaan. (Lindberg 2010, 3-9.)

Evoluutiomallin suuri suosio johtuu sen mahdollisuudesta lisätä uusia ominaisuuksia ja osia järjestelmään jo sitä tehdessä. Evoluutiomallin alussa tehdyt määrittelyt ja suunnittelut eivät ole ensisijaisia vaan suuntaa-antavia. Koska projekti rakennetaan vaiheittain, määrittelyjä voidaan tarkentaa ja ominaisuuksia lisätä jo yhden välivaiheen jälkeen. Evoluutiomalli on osittain vesiputousmallin mukainen. (Lindberg 2010, 6.)

Evoluutiomallilla on kuitenkin vahvuutensa ja heikkoutensa. Vahvuuksina voidaan mainita: nopeat tulokset, määrittelyiden lisääminen ja muuttaminen vaarantamatta projektia sekä tavoitteiden ja suuntien muuttaminen kesken projektin. Heikkouksina voidaan mainita: ensimmäisen vaiheen tulos ei ole lopullinen, pitkäjänteinen työ ja jatkuva kehittäminen, ei välttämättä valmistu uusien ideoiden ja tarpeiden tullessa esille sekä pahimmillaan vaikeasti hallittava, jatkokehittettävä, hajanainen kokonaisarkkitehtuuri. (Lindberg 2010, 10.)



Kuvio 2. Järjestelmässä käytettävä evoluutiomalli

Koska järjestelmä on moniosainen, on helpompi tehdä osa kerrallaan. Ensin luodaan etusivu pohjaksi, toiseksi tietojen syöttölomake, kolmanneksi esityslomake. Miksi näin? Koska esityslomaketta ei voi tehdä ilman tietoja, ja tietoja ei saada ilman syöttölomaketta, ja kuitenkin nämä sivut täytyy saada johonkin näkyviin.

3.2 PHP-ohjelmointikieli

PHP (rekursiivinen lyhenne sanoista PHP: Hypertext Preprocessor) on ohjelmointikieli, jota käytetään mm. Internet-palvelinympäristöissä dynaamisissa web-sivuissa. PHP:ssä on laaja luokkakirjasto. PHP luetellaan komentosarjakieleksi, jossa koodi tulkitaan vasta

ohjelmaa suoritettaessa. PHP:tä voidaan näin ollen käyttää useilla eri alustoilla ja käyttöjärjestelmillä.

PHP:n ensimmäinen versio ilmestyi vuonna 1995, ja nykyään PHP johtaa vertailua dynaamisten web-palveluiden tuottamiseen tarkoitettuna kielenä. Vuoden 1998 lopulla Andi Gutmans ja Zeev Suraski käynnistivät PHP:n ytimen uudelleen ohjelmoinnin. Tavoitteena oli saada tuki kolmansien osapuolten ohjelmointirajapinnoille. Uusi ydin ilmestyi vuonna 1999 nimellä Zend Engine ja vuonna 2000 PHP4 käytti sitä ytimenään. PHP5 ilmestyi kesällä 2004. Tuolloin ytimenä oli jo Zend Engine II, joka tuki olio-ohjelmointia ja sisälsi tietokantamoottorin (SQLite). Uusin versio 5.3.2 julkaistiin 4. päivä maaliskuuta 2010. (Wikipedia 2010a.)

Tunnettuja projekteja joihin on käytetty PHP:tä

- Wikipedia, wiki-järjestelmä
- phpBB, bulletin board-keskustelusovellus
- osCommerce, avoimen koodin verkkokauppasovellus
- Facebook, yhteisösivusto
- IRC-Galleria, yhteisösivusto
- Travian, moninpelattava selainpeli
- Moodle, avoimen koodin oppimisalusta

PHP-ohjelmointikielen valitsin siksi, että MySQL-tietokantaa on helpompi käyttää tällä kielellä ja sivuston voi tehdä dynaamiseksi. Muitakin ohjelmointikieliä olisin voinut käyttää, mutta itselläni on paljon enemmän kokemusta PHP-kielestä. Jo sivustolle kirjaututtaessa on käytössä PHP-kieltä. PHP-koodi saa käyttäjätunnuksen ja salasanan, jotka syötetään kirjautuessa, joita se vertaa tietokannasta löytyviin tileihin. Tämä jo itsessään on tietoturvallinen tapa, mutta lisäsin vielä että salasanaa verrataan md5-hashiin, joka löytyy tilien tiedoista.

Sisälle kirjaututtaessa etusivun linkit on tehty dynaamisesti käyttäen PHP:tä. Vuoden vaihtuessa tulee uusi tietojensyöttölinkki, jolla saadaan lisättyä edellisen vuoden tiedot. Joka seurantavuodelle tulee oma linkki, sitä myöten kun vuodet vaihtuvat.

Flash-kaavioissa PHP tulee esiin tiedonhaku-, tiedonkäsittely- ja tulostusvaiheissa. Tiedonhaussa käytän PHP:ta kun tiedot pitää hakea tietokannasta. Tiedonkäsittelyvaiheessa kun tietokannasta haettu data pitää muuttaa Flash-kaavion ymmärtämään muotoon, tämä onnistuu helposti kaavion mukana tulleen kirjaston avulla. Tulostusvaiheessa muotoiltu data tulostetaan JavaScriptille, joka sisällyttää sen Flash-kaavioon.

JavaScriptiä käytetään esittämään Flash-sovellus. Tämä koodi kuviossa 3 ei ole välttämätön, mutta erittäin käytännöllinen.

Johtuen monista lakiasioista, Internet Explorer-selain ei pysty "käynnistämään" Flash-kaavioita ilman että käyttäjän pitäisi aktivoida kaaviot ensin. Mutta onneksi, tämä voidaan kiertää pienellä määrällä JavaScript-koodia.

`swfobject.embedSWF(swfUrl, id, width, height, version, expressInstallSwfurl, flashvars, params, attributes, callbackFn)` sisältää viisi pakollista ja viisi valinnaista muuttujaa:

1. `swfUrl` (merkkijono, pakollinen) määrittelee suoritettavan Flash-sovellukseen URL-osoitteen.
2. `id` (merkkijono, pakollinen) määrittelee HTML-elementin id:n, joka sisältää vaihtoehtoisen sisällön, joka korvataan Flash-sovelluksella.
3. `width` (merkkijono, pakollinen) määrittelee Flash-sovelluksen leveyden
4. `height` (merkkijono, pakollinen) määrittelee Flash-sovelluksen korkeuden
5. `version` (merkkijono, pakollinen) määrittelee Flash Player version, mille Flash-sovellus on julkaistu (muoto on: "major.minor.release" tai "major")

`JSON.stringify` muuntaa JavaScriptin datarakenteet JSON-formaattiin tekstiksi.

Funktio `findSWF` käyttää ikkuna- ja dokumenttiryhmiä tunnistaaakseen Flash-sovelluksen. Adobelta on sanottu, että tämä on luotettavampi tapa laajalle joukolle selaimia kuin esimerkiksi `getElementById()` menetelmä.


```

<html>
<head>

<script type="text/javascript" src="../../js/json/json2.js"></script>
<script type="text/javascript" src="js/swfobject.js"></script>
<script type="text/javascript">
swfobject.embedSWF("open-flash-chart.swf", "my_chart", "350", "200",
"9.0.0");
</script>

<script type="text/javascript">

function open_flash_chart_data()
{
    return JSON.stringify(data);
}

function findSWF(movieName) {
    if (navigator.appName.indexOf("Microsoft") != -1) {
        return window[movieName];
    } else {
        return document[movieName];
    }
}

var data = TULOSETTAVA TIETO TULEE TÄHÄN PHP:N AVULLA;

</script>

</head>
<body>

<p>Hello World</p>

<div id="my_chart"></div>

</body>
</html>

```

Kuvio 3. JavaScriptin ja Flash-kaavion yhteys

PHP:n vahvuuksina voidaan mainita tietokantatuki mm. MySQL, PostgreSQL, SQLite ja Oracle -tietokannoille. PHP:n heikkouksina voidaan mainita vajaa Unicode -merkistö, hatarat sisäinen yhtenäisyys, ei sisäänrakennettua säikeiden käsittelyä (sivu latautuu vasta, kun koko tietomäärä on käsitelty).

3.3 MySQL-tietokanta

MySQL on SQL-tietokannan hallintajärjestelmä. MySQL:n omistaa nykyään Sun Microsystems. Monista kaupallisista versioista poiketen MySQL:n hallinnointi tapahtuu komentorivillä tai tekstipohjaisella asiakasohjelmalla. Valmistajalta löytyvät omat ohjelmat MySQL Administrator sekä MySQL Browser. Yksi suosituimmista ohjelmista on phpMyAdmin, joka on tehty PHP:n pohjalle. MySQL:ää on vierastettu ammattilaispiireissä vajaavaisuuksien takia, kuten transaktioiden ja vierasavaimien puutteen takia, mutta MySQL on saanut vajaavaisuudet korjattua ja saanut eroa kiinni.

MySQL luotiin vuonna 1995 yhdessä suomalaisen Michael ”Monty” Wideniuksen ja ruotsalaisen David Axmarkin kanssa. Ensimmäinen versio tästä tietokannasta ilmestyi vuonna 1996. Uusin ja suositeltavin on MySQL:n versio 5.1. MySQL on hyvin suosittu dynaamisten web-palveluiden tietokantana. Ohjelmalogiikka, joka rakennetaan tietokannan päälle, toteutetaan usein PHP, Python tai Perl- ohjelmointikielellä. Sivut julkaistaan Apache-webpalvelimella, joka toimii nykyään joko Windows- tai Linux-käyttöjärjestelmän päällä. Näitä kutsutaan WAMP(Windows, Apache, MySQL, PHP)- ja LAMP(Linux, Apache, MySQL, PHP)-alustoiksi. MySQL:ää voi käyttää myös muilla ohjelmointikielillä. MySQL sisältää rajapinnan mm. C:lle, C++:lle, C#:lle, Javalle ja Rubyille. MySQL:lle on tehty myös MyODBC-niminen ODBC-rajapinta. (Wikipedia 2010b.)

Sivustot, jotka käyttävät MySQL-tietokantaa

- Wikipedia
- Google
- Yahoo!

Tietokanta koostuu tauluista, jotka taas koostuvat kentistä. Taulut ovat kuin normaaleja taulukkoja. Niissä on rivit ja sarakkeet. Sarake sisältää yhden kentän ja yksi rivi on tietue. Taulu voi sisältää kenttiä muutamasta kappaleesta satoihin. Kenttä määritellään taulua luotaessa. Kenttiä voi olla monia erilaisia. Osa kentistä voi hyväksyä vain luvun, tekstin, totuusarvon, tai jonkin muun tallennettavan tyyppisen määritelmän. Kentille määritellään myös enimmäispituus. Kentistä yksi on avainkenttä eli perusavain yksilöi-

mään tietueen. Esimerkiksi jos kaikki suomalaiset laitetaan taulukkoon. Mikä erottaisi Matti Juhani Meikäläiset toisistaan? Sosiaalityrvtunnus olisi hyvä avain, koska ei ole olemassa kahta samanlaista tunnusta.

Taulujen välille täytyy luoda yhteyksiä, jotta tietoja voidaan hakea useista tauluista niiden sekoamatta muiden tietoihin. Esimerkiksi kirjastossa asiakkailla on omat tunnukset ja kirjoilla on omat tunnukset. Kirjoja lainatessa kirjojen tunnukset linkitetään lainaajan tunnukseen. Ajattele, jos linkitys ei menekään oikein, lainaamasi kirjat näkyvät muilla asiakkailla ja sinä saat itsellesi muiden lainauksia.

Kun MySQL -tietokannasta halutaan kyselyiden avulla haluttua tietoa, voidaan hakea tietoa useista eri tauluista. Kyselyt noudattavat aina tiettyä rakennetta.

1. **SELECT** taulun_nimi.kentän_nimi (mitkä kentät halutaan mukaan)
2. **FROM** taulu, taulut (mistä taulusta / tauluista kentät ovat)
3. **WHERE** (miten tietoa suodatetaan (ehdot))
4. **GROUP BY** (miten ryhmitellään)
5. **HAVING** (ryhmittelyn ehdot)
6. **ORDER BY** (miten lajitellaan)

Pakolliset osat ja yksinkertainen esimerkki:

```
SELECT etunimi, sukunimi FROM oppilas
```

Taulukossa 1 on esitetty esimerkki tietokantataulukosta, jossa on kaksi kenttää: etunimi ja sukunimi. Jokainen etunimi/sukunimi -rivi on yksi tietue.

Taulukko 1. Esimerkki MySQL:n käytöstä

etunimi	sukunimi
Matti	Meikäläinen
Matti	Mainio
Jussi	Juonio
Maija	Meikäläinen

3.4 JSON-formaatti

JSON (JavaScript Object Notation) on kevyt tiedonsiirtoformaatti. Sitä on helppo lukea ja kirjoittaa, niin ihmisten kuin koneidenkin. JSON on tekstiformaatti joka on kokonaan kielestä riippumaton, mutta käyttää tapaa joka on tuttu ohjelmoijille C-perheen kielistä (C, C++, C#, Java, JavaScript, Perl, Python). Nämä ominaisuudet tekevät JSON-formaatista ideaalisen tiedonsiirtokielen. (JSON.)

JSON on rakennettu kahdelle rakenteelle:

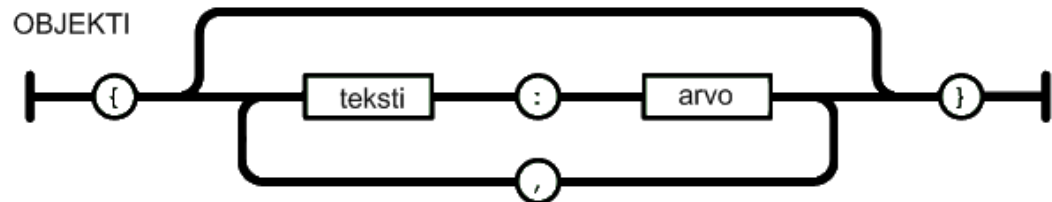
- Kokoelma nimi/arvo-pareja. Useissa ohjelmointikielissä, tämä on ymmärretty objektina, tallenteena, tietueena, sanakirjana, taulukkona, avainlistana tai assosiatiivisena taulukkona.
- Järjestetty arvolista. Useissa ohjelmointikielissä, tämä on ymmärretty taulukkona, vektorina, listana tai sarjana.

(JSON.)

Nämä ovat yleisiä tietorakenteita. Virtuaalisesti kaikki modernit ohjelmointikielet tukevat näitä tavalla tai toisella. On selvää, että tietformaatti, joka on siirrettävissä ohjelmointikielissä, perustuu näille rakenteille.

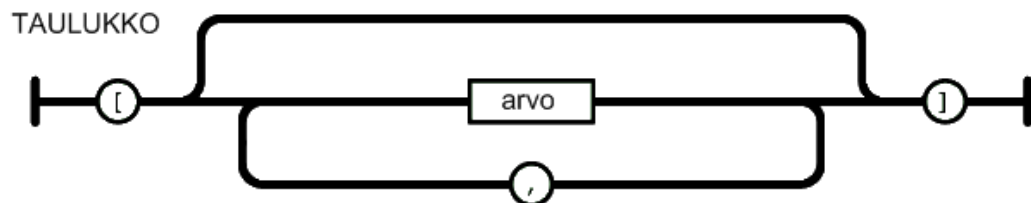
JSON-formaatissa nämä rakenteet näkyvät tällä tavalla:

- **Objekti (object)** on järjestelemätön joukko nimi/arvo-pareja. Objekti alkaa ja loppuu aaltosululla ({ }). Jokaisen nimen perään lisätään kaksoispiste (:) ja nimi/arvo-parit erotetaan toisistaan pilkulla (,).



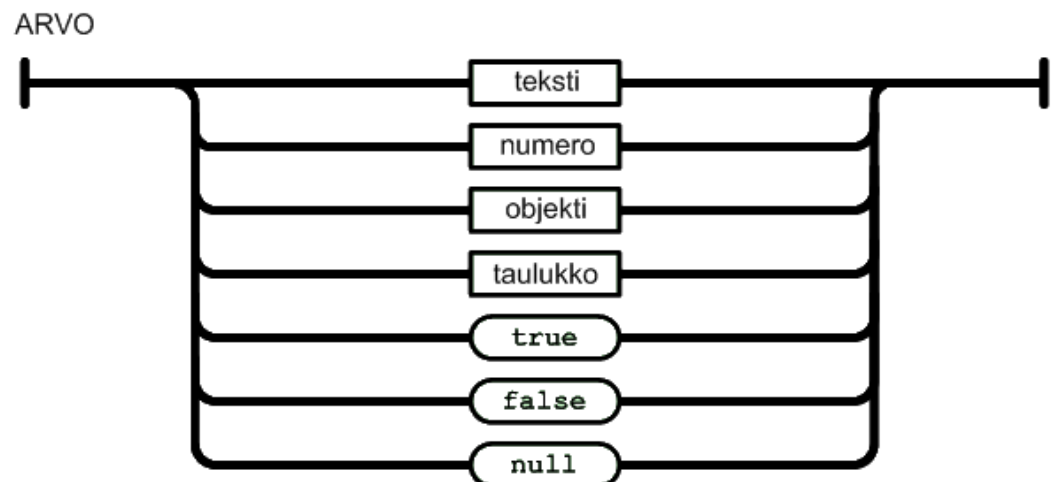
Kuvio 4. Objekti

- **Taulukko (array)** on järjestetty kokoelma arvoja. Taulukko alkaa ja loppuu hakusululla ([]). Arvot erotetaan toisistaan pilkulla (,).



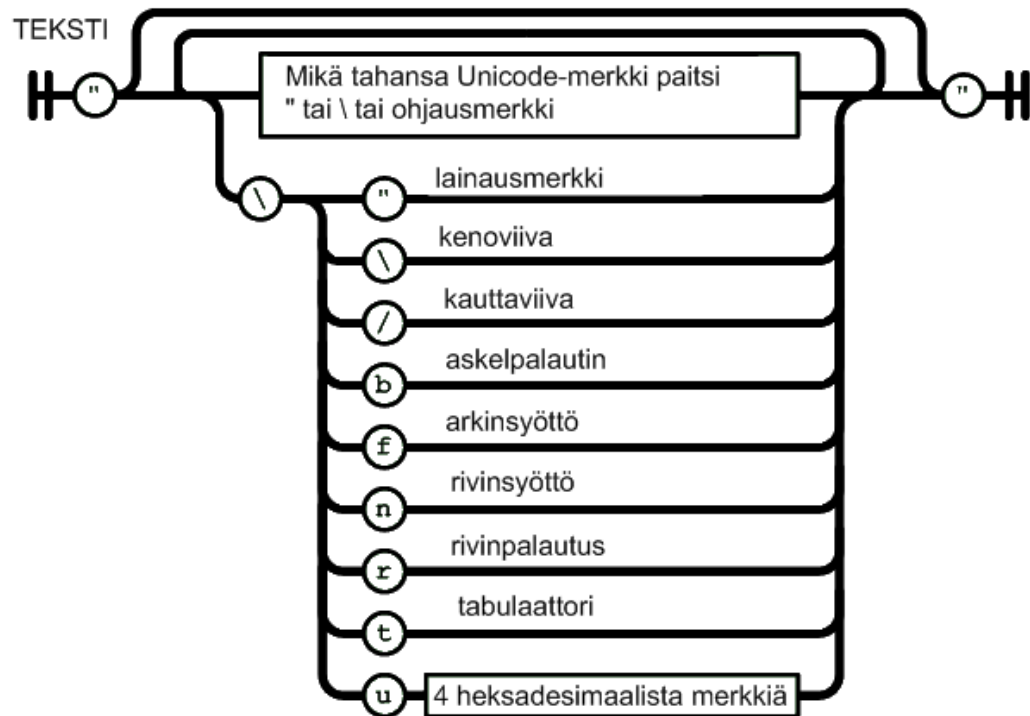
Kuvio 5. Taulukko

- **Arvo (value)** voi olla teksti lainausmerkeissä, numero, tosi, epätosi, null(tyhjä arvo), objekti tai taulukko. Nämä rakenteet voidaan sijoittaa sisäkkäin.



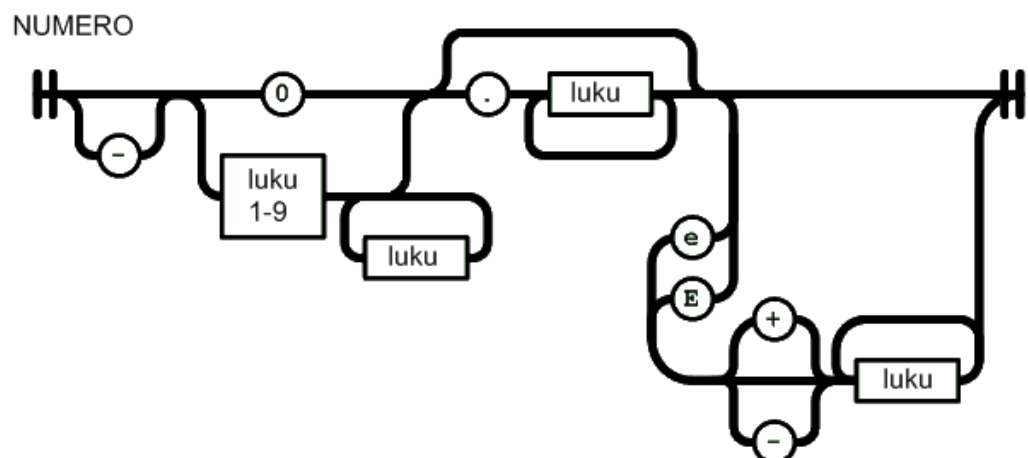
Kuvio 6. Arvo

- **Teksti (string)** on kokoelma nollia tai muita Unicode-merkistöstandardin merkejä, käärittynä lainausmerkkeihin, käyttäen kenoviivaa ohjausmerkkeihin. Merkki esitetään yksittäisenä merkkijonona. Teksti on samanlainen kuin C- tai Java-kielen teksti.



Kuvio 7. Teksti

- **Numero** on samanlainen kuin C- tai Java-kielen numero, paitsi että oktaali- ja heksadesimaalijärjestelmiä ei käytetä.



Kuvio 8. Numero

Tyhjiä välilyöntejä voidaan lisätä mihin tahansa merkkien väliin. Paitsi muutamaa yksityiskohtaa koodauksessa, jotka lopullisesti määrittelevät kielen.

JSON on yksinkertainen ja selkeä. JSON koostuu nimi/arvo -pareista.

```
{
  "elements": [
    {
      "type": "bar_glass",
      "colour": "#9933CC",
      "text": "Omat tiedot",
      "values": [
        27, 18, 10, 25, 16, 13, 20
      ]
    },
    {
      "type": "line_dot",
      "colour": "#CC3399",
      "text": "Toimiala keskiarvo",
      "values": [
        17, 20, 18, 16, 15, 18, 19
      ]
    }
  ],
  "title": {
    "text": "Henkilöstön käyttö (henkilötyövuotta)"
  },
  "x_axis": {
    "labels": {
      "labels": [
        "2007",
        "2008",
        "2009",
        "2010",
        "2011",
        "2012",
        "2013"
      ]
    }
  }
}
```

```

},
"y_axis": {
  "min": 0,
  "max": 28,
  "steps": 1
}
}

```

Kuvio 9. Esimerkki JSON- formaatista

Kuviossa 9 on esitetty esimerkki JSON-formaatista kahdella eri arvojoukolla. Formaattissa on useita eri elementtejä, kuitenkin seuraavat elementit ovat vaadittuja: tyyppi, arvot, y-akselin arvot. Edellä mainitussa esimerkissä ensimmäisenä on määritetty ensimmäisen arvojoukon näyttötyyppi "bar_glass" (palkki_lasinen) ja arvot 27, 18, 10, 25, 16, 13, 20. Toisena on määritelty seuraavan arvojoukon näyttötyyppi "line_dot" (viiva_pisteellä) ja arvot 17, 20, 18, 16, 15, 18, 19. Seuraavaksi annetaan kaaviolle otsikko "Henkilöstön käyttö (henkilötyövuotta)", x-akselin otsikot 2007-2013 ja y-akselin arvot (min)minimi:0, (max)maksimi:28 ja (step)askellus:1. Näistä elementeistä ja arvoista syntyy tällainen kaavio.



Kuvio 10. Kuvio 9 luotu kaavio

3.5 Graafisia esityksiä Internetissä - Flash-kaaviot

Flash-kaaviot ovat PHP-kieleen laadittuja lisäpalveluita, joiden avulla voidaan luoda dynaamisia ja interaktiivisia graafisia esityksiä Internet-sovelluksiin.

Flash-kaaviot auttavat luomaan animoituja ja interaktiivisia kaavioita Internet- ja työ-pöytäsovelluksiin. Nämä kaaviot tuovat eloa sovelluksiin muuttamalla monotonisen datan kiinnostavaksi visuaaliseksi kaavioksi. (FusionCharts.)

Lyhyt esitys siitä mitä tapahtuu:

1. Selain pyytää sivua chart.html
2. Selain löytää Flash tagit ja pyytää open-flash-chart.swf Flash-sovellusta. Selain lataa sovelluksen.
3. Open Flash Chart etsii URL:n avulla data-tiedoston. Lataa sen.
4. Open Flash Chart lukee JSON-tiedon ja näyttää graafisen kaavion.

(Open Flash Chart 2 tutorial.)

Flash-kaaviot on toteutettu niin, että haluttu data voidaan syöttää pelkästä tekstitiedostosta esitettäväksi. Seuraavassa esitän miten haetaan tulostettavat tiedot tietokannasta, muutetaan ne oikeaan muotoon Flash-kaaviolle ja tulostetaan ne.

Open Flash Chart -komponentti on rakennettu käyttäen Adobe Flexiä. Flex on yhdistäjä, joka ottaa ActionScript 3 -lähdekoodin ja muuttaa sen Flash-sovellukseksi.

3.5.1 JSON-formaatti tiedonlähde Flash-kaaviolle

Open Flash Chartin uudempi versio käyttää tiedostomuotona JSON-formaattia. JSON muodostuu arvopareista. Seuraavassa on vähän esimerkkejä mitä osia siihen tarvitaan:

Title (otsikko, valinnainen)

text: merkkijono, otsikko

style: merkkijono, CSS-tyyli

esim.

```
{
  "title":{
    "text": "Esimerkki",
    "style": "{ font-size: 30px; color:#ffffff; font-family: Verdana; text-align: left;}"
  }
}
```

Y Legend (y-akselin otsikko, valinnainen)

text: merkkijono, otsikko y-akselille

style: merkkijono, CSS-tyyli

esim.

```
{
  "y_legend":{
    "text": "Kaavio",
    "style": "{ color: #00FF00; font-size: 10px;}"
  }
}
```

X Axis (x-akselin arvot, valinnainen)

stroke: numero, viivan leveys

tick_height: numero, askellusmerkkien korkeus

colour: merkkijono, viivan väri

grid_colour: merkkijono, taulukkoristikon väri

3d: totuusarvo, onko 3d vai ei

steps: näyttää joka n:n askelluksen

labels: merkkijonotaulukko, otsikot jokaiselle x-akselin pisteelle

esim.

```
{
  "x_axis":{
    "stroke": 2,
    "tick_height": 10,
    "colour": "#0d000d",
```

```

    "grid_colour": "#ff00ff",
    "labels": ["Tammi-
kuu", "Helmikuu", "Maaliskuu", "Huhtikuu", "Toukokuu", "Kesäkuu", "Heinäkuu", "Elokuu", "Syyskuu"]
  }
}

```

Y Axis (y-akselin arvot, valinnainen)

Muuten samat kuin x-akselilla muutama lisää

offset: numero, aloitus

max: numero, maksimi

esim.

```

{
  "y_axis": {
    "stroke": 3,
    "tick_length": 4,
    "colour": "#0d000d",
    "grid_colour": "#ff00ff",
    "offset": 0,
    "max": 30
  }
}

```

Elements

Elementtimäärite sisältää taulukollisen objekteja ja jokainen objekti sisältää yhden taulukon tiedot (tyypin, värin, tekstin)

esim.

```

{
  "elements": [
    {
      "type": "bar",
      "alpha": 0.3,
      "colour": "#8833DD",

```

```

    "text":    "Sivunäkymä",
    "font-size": 10,
    "values" : [9,6,7,9,5,7,6,9,7]
  }
]
}

```

Elements.bar

type: pitää olla ”bar”, palkkikaavio
 alpha: desimaali, läpinäkyvyys
 colour: merkkijono, elementin väri
 text: merkkijono, elementin otsikko
 font-size: numero, fontin koko
 values: numerotaulukko, jokaisen palkin korkeusarvot

esim.

```

{
  "elements":[
    {
      "type":    "bar",
      "alpha":   0.4,
      "colour":  "#8833DD",
      "text":    "Palkki",
      "font-size": 9,
      "values" : [9,6,7,9,5,7,6,9,7]
    }
  ]
}

```

Elements.pie

type: pitää olla ”pie”, piirakkakaavio
 start-angle: numero, kulma ensimmäiselle piirakanpalalle
 colours: merkkijonotaulukko, CSS-väri, piirakanpalojen väri
 alpha: desimaali, läpinäkyvyys

stroke: numero, ulkolinjan leveys

animate: totuusarvo, animoi piirakkakaavion

values: numerotaulukko, jokaisen piirakanpalan arvot. Voi olla numero tai osaobjekti

esim.

```
{
  "elements": [
    {
      "type": "pie",
      "start-angle": 90,
      "colours": ["#d01f3c", "#356aa0", "#C79810", "#73880A", "#D15600", "#6BBA70"],
      "alpha": 0.5,
      "stroke": 2,
      "animate": 1,
      "values": [0, 2, {"value": 0, "text": "zero"}, 2, 6]
    }
  ]
}
```

Elements.hbar

type: pitää olla "hbar", vaakapalkkikaavio

colour: merkkijono, elementin väri

text: merkkijono, elementin otsikko

font-size: numero, fontin koko

values: objektitaulukko, jokaisella arvolla pitää olla "right" ja valinnainen "left" arvo

esim.

```
{
  "elements": [
    {
      "type": "hbar",
      "colour": "#883366",
      "text": "Vaakapalkki",

```

```

    "font-size": 9,
    "values" : [{"right":9},{ "right":16},{ "left":18,"right":19}]
  }
]
}

```

Elements.line_dot

type: pitää olla "line_dot", viivapistekaavio

colour: merkkijono, elementin väri

text: merkkijono, elementin otsikko

font-size: numero, fontin koko

width: numero, viivan paksuus

dot-size: numero, pisteen koko

values: desimaalitaulukko, jokaisen viivan pisteen korkeusarvot

esim.

```

{
  "elements":[
    {
      "type":    "line_dot",
      "colour":  "#858BFF",
      "text":    "Viivapiste",
      "font-size": 9,
      "width":   1,
      "dot-size": 3,
      "values" : [1.5,1.69,1.88,2.06,2.21,2.34,null,2.35,2.23,2.08]
    }
  ]
}

```

Elements.line*

Kolme erityyppistä

line: viiva

line_dot: viiva pisteillä

line_hollow: viiva ontoilla pisteillä

Esimerkki:

```
{
  "title":{
    "text":"Kolme eri viivaa",
    "style":"{font-size: 25px;}"
  },

  "y_legend":{
    "text":"Viivat",
    "style":"{font-size: 10px; color:#736AFF;}"
  },

  "elements":[
    {
      "type":    "line",
      "colour":  "#9933CC",
      "text":    "Viiva",
      "width":   1,
      "font-size": 10,
      "dot-size": 4,
      "values" : [15,18,19,14,17,18,15,18,17]
    },
    {
      "type":    "line_dot",
      "colour":  "#CC3399",
      "width":   2,
      "text":    "Viivapiste",
      "font-size": 10,
      "dot-size": 3,
      "values" : [10,12,14,9,12,13,10,13,12]
    },
  ],
}
```

```

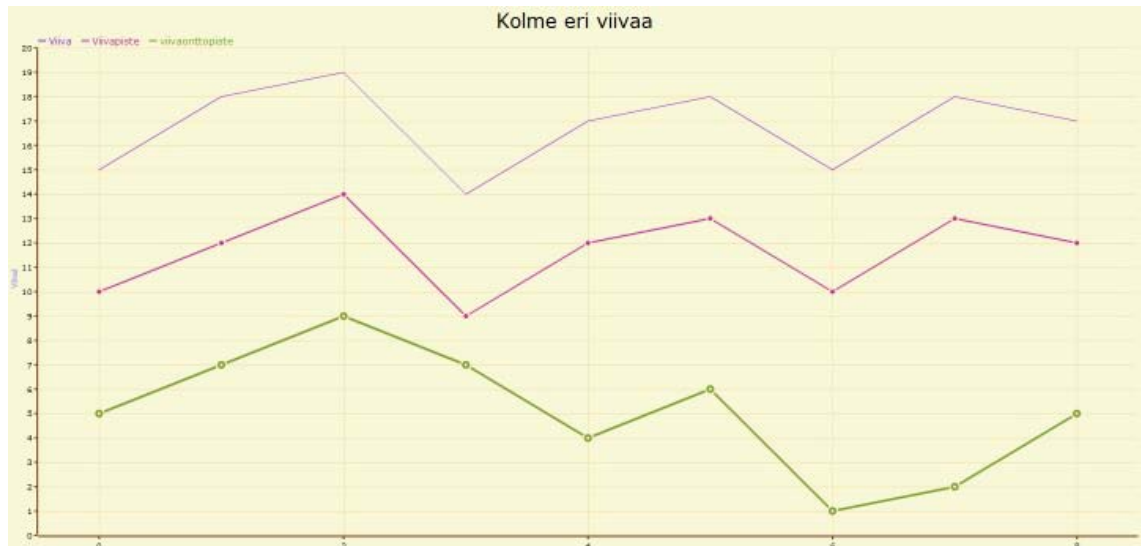
{
  "type": "line_hollow",
  "colour": "#80a033",
  "width": 3,
  "text": "viivaonttopiste",
  "font-size": 10,
  "dot-size": 5,
  "values": [5,7,9,7,4,6,1,2,5]
}
],

"y_axis":{
  "max": 20
},

"x_axis":{
  "steps": 2,
  "labels": ["Tammi", "Helmi", "Maalis", "Huhti", "Touko", "Kesä", "Heinä", "Elo", "Syys"]
}
}

```

Kuviossa 11 nähdään edellä mainitun arvosarjan lopputulos.



Kuvio 11. Kolme eri viivaa

Seuraavaksi näytän kuinka nämä arvot saadaan esitettyä Flash-kaaviona

3.5.2 Esimerkki Flash-kaaviosta

Seuraavalla yksinkertaisella esimerkillä yritän havainnollistaa, miten Flash-kaavio toimii.

```

<html>
<head>
</head>
<body>

<p>Hello World</p>

<object classid="clsid:d27cdb6e-ae6d-11cf-96b8-444553540000"
code-
base="http://fpdownload.macromedia.com/pub/shockwave/cabs/flash/swflash.cab#version=8,0,0,0"
width="500"
height="250" id="graph-2" align="middle">

    <param name="allowScriptAccess" value="sameDomain" />
    <param name="movie" value="open-flash-chart.swf" />
    <param name="quality" value="high" />
    <embed src="open-flash-chart.swf"
quality="high"
bgcolor="#FFFFFF"
width="500"
height="250"
name="open-flash-chart"
align="middle"
allowScriptAccess="sameDomain"
type="application/x-shockwave-flash"
pluginspage="http://www.macromedia.com/go/getflashplayer" />
</object>

</body>
</html>

```

Kuvio 12. Yksinkertainen esimerkki html-koodista

Tämä koodi laitetaan tavallisen html-sivun lähdekoodiin (esim. chart.html). Koodissa käytetään sekä object- että embed-tageja, koska Internet Explorer -selain käyttää yhtä tapaa ja loput selaimet käyttävät toista. Kun sivu avataan, aluksi näkyy teksti ”Hello World” ja vähän ajan kuluttua tulee näkyviin virheilmoitus. (Kuvio 13.)

```

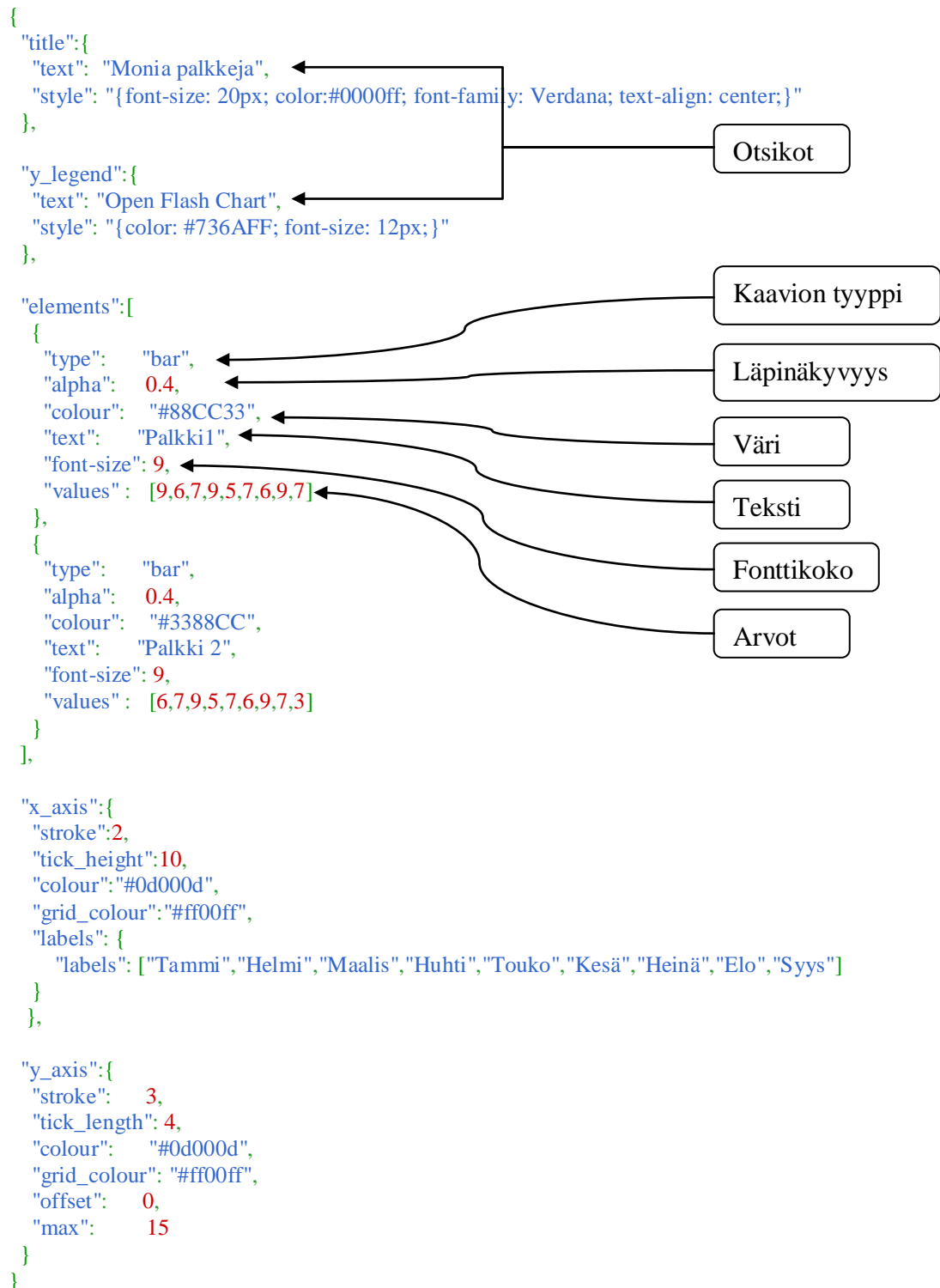
IO ERROR
Loading test data
Error #2032

```

Kuvio 13. Virheilmoitus

Open Flash Chart ei löytänyt datatiedostoa. Lisätään sille sitten sellainen.

Liitetään seuraava arvosarja uuteen tiedostoon data.json



Kuvio 14. Esimerkki Flash-kaavion luovasta koodista

Kun nyt lisäämme URL-polun loppuun `?ofc=data.json`, saamme kaipaamamme kaavion.

Open Flash Chart etsii muuttujaa ofc URL-polusta, jolle annetaan osoite data-tiedostoon.



Kuvio 15. Flash-kaavio

3.5.3 Esimerkki järjestelmän Flash-kaaviosta

Tässä on kuvattu eräs järjestelmässä käytetyn kaavion toimintaidea – yritysten liikevaihto (tuhansina euroina).

Alkutiedot: muuttuja \$q (question) saa arvon lomakkeen valikosta, joka on 2 (eli kysymys 2) ja vuodeksi on valittu 2007.

MySQL -tietokannasta haemme PHP:n avulla tarvittavat tiedot:

```
/** taulukon maksimiarvo */
$max="SELECT max(answer".$q.") FROM answers";
$max_value=mysql_query($max)or die('@ic1'.mysql_error());
$max_value_result=mysql_fetch_array($max_value)or die('@ic2'.mysql_error());
$max_int=$max_value_result[0];

/** taulukon stepping */
$step_int=round($max_value_result[0]/10,0);
```

Kuvio 16. Järjestelmän koodi osa 1

Rivi 1: Tehdään SQL-kysely, joka hakee taulukon suurimman mahdollisen arvon muuttujan \$q mukaan kaikista kyseisistä annetuista vastauksista.

Rivi 2: Suoritetaan kysely MySQL-tietokannasta tai tiedotetaan virheellä jos näin tapahtuu.

Rivi 3: Noudetaan tulos taulukosta

Rivi 4: Tallennetaan tulos \$max_int muuttujaan.

Rivi 5: Graafisen kuvaajan askellus on kymmenys juuri saadusta tuloksesta, pyöristettynä.

```
/** tästä alkaa omien vastausten nouto **/
$my_sql="SELECT answer".$q." FROM answers WHERE yritys_id = ".$_SESSION['memberID']."
AND vuodenXluvut = ".$_GET['vuosi']."";

$my_result = mysql_query($my_sql) or die('@ic3'.mysql_error());

while($row = mysql_fetch_array($my_result))
{
    $my_data[]=intval($row[0]);
}
```

Kuvio 17. Järjestelmän koodi osa 2

Rivi 6: Tehdään SQL-kysely, joka hakee yrityksen omat tiedot kyseiseltä vuodelta. Kyselyn muuttujat \$q, \$_SESSION['memberID'] ja \$_GET['vuosi'] saavat arvonsa lomakkeen tiedoista sekä kirjautumistiedoista.

Rivi 7: Suoritetaan kysely tai tulostetaan virhe.

Rivi 8: Tehdään silmukka, joka lisää jokaisen saadun tuloksen yksitellen uuteen taulukon solu kerrallaan.

```

/** tästä alkaa muiden vastausten nouto */

/** haetaan nykyisen käyttäjän toimiala */
$toimi_sql="SELECT answer3 FROM answers WHERE yritys_id=".$$_SESSION['memberID']."";
$toimi_value=mysql_query($toimi_sql)or die('@ic4'.mysql_error());
$toimi_value_result=mysql_fetch_array($toimi_value)or die('@ic5'.mysql_error());
$toimi=$toimi_value_result[0];

/** tästä alkaa vastausten nouto*/
$other_sql="SELECT round(avg(answer".$q.",0) FROM answers WHERE vuodenXluvut = ".$$_GET['vuosi']. " AND answer3='".$toimi.'";

$other_result = mysql_query($other_sql)or die('ic6'.mysql_error());

while($row = mysql_fetch_array($other_result))
{
    $other_data[]=intval($row[0]);
}

```

Kuvio 18. Järjestelmän koodi osa 3

Rivi 9: Tehdään SQL-kysely, jolla selvitetään nykyisen käyttäjän toimiala.

Rivi 10: Suoritetaan kysely tietokannasta tai tulostetaan virhe.

Rivi 11: Noudetaan tiedot taulukkoon.

Rivi 12: Otetaan ensimmäinen solu taulukosta käyttöön, joka on käyttäjän toimiala-arvo ja tallennetaan toimi-muuttujaan.

Rivi 13: Tehdään SQL-kysely, jolla haetaan saman toimialan yritysten keskiarvo tietyin vuosi ja toimi-muuttujiin

Rivi 14: Suoritetaan kysely

Rivi 15: Tehdään silmukka, joka lisää jokaisen saadun tuloksen yksitellen uuteen taulukkoon solu kerrallaan.

Seuraavaksi syötämme yllä haetut tiedot PHP:lla taulukkoon tietyin nimi ja arvoparein:

```

include '../php-ofc-library/open-flash-chart.php';

$title = new title('Tulokset');

$bar = new bar_glass();
$bar->set_values( $other_data );//query
$bar->set_colour( '#770000' );
$bar->set_key( 'Toimiala keskiarvo', 12 );

$line = new line_dot();
$line->set_values( $my_data );//query
$line->set_colour( '#000077' );
$line->set_key( 'Omat tiedot', 12 );

$x = new x_axis();
$x->set_labels_from_array(
    array( $_GET['vuosi'] )
);

$y = new y_axis();
$y->set_range(0,$max_int,$step_int);

$chart = new open_flash_chart();
$chart->set_title( $title );
$chart->add_element( $bar );
$chart->add_element( $line );
$chart->set_x_axis( $x );
$chart->set_y_axis( $y );

```

Kuvio 19. Järjestelmän koodi osa 4

Rivi 1: Otetaan käyttöön Open Flash Chart library for PHP -kirjasto, joka löytyy Open Flash Chart -sivustolta

Rivi 2: Taulukon otsikoksi (title) on määriteltä tässä tapauksessa Tulokset.

Rivit 3-6: Käytetään kirjaston objektia bar_glass (palkki_lasi) muuttujaan bar. Annetaan muuttujalle taulukon arvot (toimialan keskiarvo), väri (heksamuodossa) ja palkin otsikko sekä tekstin koko.

Rivit 7-10: Samanlainen muuten kuin palkin kanssa mutta käytetään objektia line_dot (viiva_piste) muuttujaan line. Objektin taulukon arvot ovat tällä kertaa omat tulokset.

Rivit 11-12: Käytetään elementtiä x_axis eli x-askelisto. Annetaan muuttujalle vielä x-akselin arvot (otsikot), tässä tapauksessa käytetään pelkästään tarkasteltavaa vuotta.

Rivit 13-14: Käytetään elementtiä `y_axis` eli y-akselisto. Annetaan muuttujalle y-akselin kantama-arvot eli minimi-, maksimi- ja askellusarvot.

Rivit: 15-20: Käytetään objektia `open_flash_chart` eli kaikki edellä tehdyt kerätään tänne esitystä varten. Elementin arvoiksi annetaan title (otsikko), elementtimuuttujat bar ja line, x- ja y-akselit.

Tähän loppui tietojen kerääminen ja tallentaminen esitettävään muotoon. Enää jäljellä on tietojen esittäminen Flash-kaaviossa.

Kaikki tähän saakka tehty koodi on käyttäjiltä piilossa, seuraava vaihe tulee näkyviin sivuston html-koodissa.

Seuraava koodi liittää aikaisemmin noudetut tiedot Flash-kaavioon JavaScriptin avulla:

```
<script type="text/javascript">

function ofc_ready()
{

}

function open_flash_chart_data()
{

    return JSON.stringify(data);
}

function findSWF(movieName) {
    if (navigator.appName.indexOf("Microsoft") != -1) {
        return window[movieName];
    } else {
        return document[movieName];
    }
}

var data = {
    "elements": [
        {
            "type": "hollow_dot",
            "values": [
                300
            ],
            "colour": "#770000",
            "text": "Toimiala keskiarvo",
            "font-size": 12
        },
        {
            "type": "dot",
```



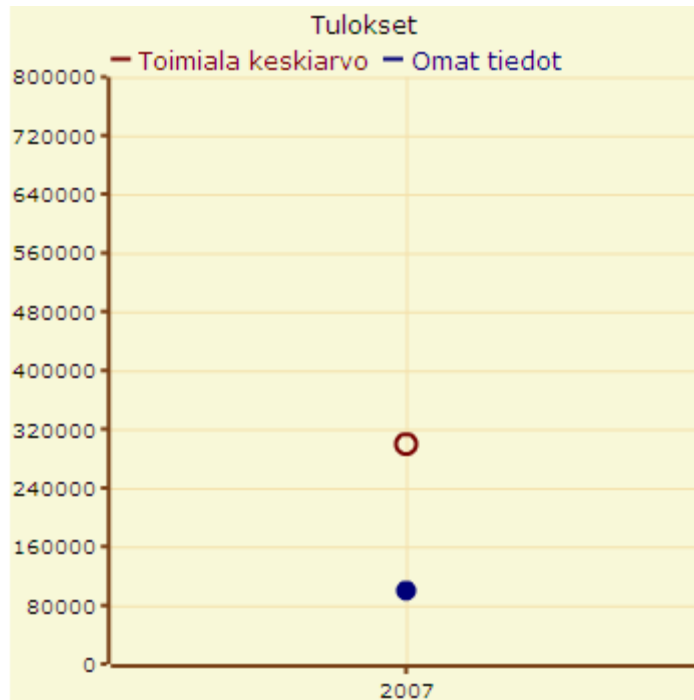
```

    "values": [
      100
    ],
    "colour": "#000077",
    "text": "Omat tiedot",
    "font-size": 12
  }
],
"title": {
  "text": "Tulokset"
},
"x_axis": {
  "labels": {
    "labels": [
      "2007"
    ]
  }
},
"y_axis": {
  "min": 0,
  "max": "800",
  "steps": 80
}
}
</script>

```

Kuvio 20. Järjestelmän kaavion ”tulostus”

Kun nämä kaikki saadaan toimimaan yhteen, saadaan seuraavanlainen kaavio (kuvio 21).



Kuvio 21. Järjestelmän kaavio

4 SEURANTAJÄRJESTELMÄ

4.1 Määrittely

Seurantajärjestelmä on osa ohjelmaa jolla ohjataan eri toimialojen yritysten kehittämis- ja investointirahoitusta. Seurantajärjestelmällä pyritään seuraamaan eri toimialayritysten kuluja, joita on käytetty erinäisiin prosesseihin. Prosesseja ovat asiakas-, teknologia-, liiketoiminta- ja talousprosessi.

1. Asiakasprosessi sisältää asiakkaisiin liittyvät kysymykset.
2. Teknologiaprosessi sisältää hankintaan liittyvät kysymykset.
3. Liiketoiminta sisältää tuotteisiin, markkinointiin, hankintaan, koulutukseen ja henkilöstöön liittyvät kysymykset.
4. Talousprosessi sisältää vientiin, markkinointiin ja pääomaan liittyvät kysymykset.

Tavoite on saada yritysten tiedot järjestelmään ja käsitellä niitä niin että ne ovat ymmärrettäviä graafisia taulukoita monotonisten arvotaulukoiden sijaan.

Tiedot kerätään syöttölomakkeelle, jossa on useita kysymyksiä toiminnan eri osalueilta (Liite 1). Kun tiedot on saatu onnistuneesti syötettyä syöttölomakkeen kautta järjestelmään, voidaan niitä alkaa käsittelemään keskenään. Voidaan verrata yrityksen omia tuloksia saman toimialan yritysten syöttämiin arvoihin keskiarvoina.

4.2 Suunnittelu

Seurantajärjestelmää varten tarvitaan tietojen syöttölomake. Lomakemalli saatiin projektista valmiina (Liite 1). Toteutin sen selaimella käytettäväksi muutaman sivun mittaiseksi lomakkeeksi.

Syöttölomakkeella kysytään millaisia arvoja kuhunkin prosessiin on mennyt kyseisenä vuonna. Asiakasprosessi sisältää kysymyksiä asiakasmääristä laskutuksessa, tyytyväisyys- ja markkinatutkimusten lukumäärästä ja reklamaatioiden lukumäärästä.

Kuisman valmiiksi tekemästä lomakkeesta tein järjestelmään oman version, joka kuitenkin vastaa suurilta osin Kuisman tekemää mallia. Oma versioni on jaettu osiin:

1. perustiedot
2. asiakasprosessi
3. teknologiaprosessi
4. liiketoimintaprosessi
5. talousprosessi

Jokainen näistä osista on omalla sivullaan. Jokaisen osion jälkeen tiedot tallentuvat järjestelmään. Viimeisen osion jälkeen tiedot tallentuvat järjestelmään ja syöttölomake palautuu kiitoksen kautta etusivulle. Tämän jälkeen tiedot ovat valmiina vertailtaviksi. Toisin kuin henkilötyötunteina tämä veisi useita tunteja, tämä vie vain muutamia sekunteja.

4.3 Toteutus

Sivut on toteutettu käyttäen PHP-ohjelmointikieltä ja pohjana käytetään MySQL-tietokantaa. Kaavion tekoon otin käyttöön Flash-kaavion, johon voidaan syöttää tietoa dynaamisesti, tieto voidaan hakea tietokannasta suoraan ja näin tulostaa kaavio.

Sivusto vaatii kirjautumisen ennen kuin voidaan syöttää tietoja tai katsella kaavioita.

4.4 Toteutustapa

Tarkoitus on toteuttaa luotavan palvelun eri osat yksi kerrallaan ja näin edeten kokonainen järjestelmä. Ensin pitää tehdä sivustolle etusivu, pohja, jonka päälle muut osat voi rakentaa.

Etusivu on tehty käyttäen normaalia taulukkoa sivun pohjalla. Tämän tyyppiseen sivuun on helppo lisätä osia sivun muodon siitä kärsimättä. Etusivu sisältää linkit muihin osiin: tietojen syöttöön, tietojen tarkasteluun ja graafisiin kaavioihin.

Tietojen syöttölomake on 5-sivuinen sivusto, jossa kysymykset on jaettu omiin osiinsa. Jokaisella sivulla kysymykset on tehty taulukkoon ja sivun alareunassa on linkki seuraavalle sivulle.

Kaavioille on oma sivunsa. Kaikkiin muihin sivuihin vedoten on tämän sivun pohja tehty myös taulukolle. Sivut sisältää ylhäällä olevat linkit, joka vuodelle mitä on tutkittu, jotka avaavat kyseisen vuoden kaavion valitusta kysymyksestä. Alempana on valintalista mistä valitaan kysymys, joka näytetään hetken kuluttua alempana. Vasemmalla reunalla on linkit omiin tietoihin siltä vuodelta.

Koko sivusto on toteutettu käyttäen pelkästään HTML-, PHP-, JavaScript- ja Flash-koodausta. Kaikki muut osat sivustosta ovat itse tekemiäni, paitsi Flash-kaavioiden tekoon käytetty sovellus on saatu eräältä tuhtuneelta ohjelmoijalta. Kyseinen henkilö osti vastaavanlaisen kaupallisen Flash-kaaviosovelluksen ja ilmoitti siinä olevasta haitallisesta virheestä, jota ei lopulta koskaan korjattu. Joten kyseinen henkilö mietti että hän

osaa itse ohjelmoida samantyyppisen sovelluksen ja julkaisi oman tuotoksensa avoimen lähdekoodin mukaisena.

Sivuston teossa on käytetty vain yhtä ohjelmaa: phpDesigner. Seuraavassa muutama kommentti, miten phpDesignerin oma sivusto kertoo ohjelmistosta.

Tehokas PHP IDE ja PHP editori sisäänrakennetulla HTML, CSS ja JavaScript editorilla. Nopea. Tehokas. Aikaa säästävä! phpDesigner on enemmän kuin tehokas ja salaman nopea PHP IDE ja PHP EDITORI - se on myös täystoiminnallinen HTML-, CSS- ja JavaScript editori lisättynä toiminnoilla jotta saat työsi tehtyä – sekä aloitteleville että ammattilaiskehittäjille! phpDesigner avustaa sinua kaikessa editoinnista, analysoinnista, virheidenkorjauksesta websivustojen julkaisuun, jotka toimivat PHP, HTML, CSS ja JavaScript -koodeilla ja monilla muilla ohjelmointikielillä! phpDesigner ei pelkästään toimi suosikki PHP- ohjelmointiympäristössäsi mutta myös suosituilla JavaScript-ohjelmointiympäristöillä kuten jQuery, Ext JS, YUI, Dojo, MooTools ja Prototype! (Pham, Michael)

Flash-kaavioiden toteutuksessa on käytetty teethgrinder.co.uk -sivuston Open Flash Chart 2 -projektia. Lisää tietoa itse tiedon näyttämisestä Flash-kaaviossa myöhemmin työn loppupuolella.

4.4.1 Tietokanta

Järjestelmän tietokanta on toteutettu käyttäen MySQL-tietokantaa sen helppokäyttöisyyden ja avoimen koodin vuoksi. Taulukossa 2 on esitetty tietokantakuvaus answer-tilin taulukosta eli minkä tyyppisiä tietoja voidaan tallettaa millekin kentälle. Answer-tilin taulukko sisältää yksilöivän vastaus-id tunnisteen, yrityksen id-tunnuksen, aikaleiman (koska tiedot on annettu), vuositunnisteen (minkä vuoden luvut on tallennettu) ja vastaukset lomakkeen kysymyksiin.

Taulukko 2. Answer-tilin taulukko

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
id	int(5)	Ei	K	
yritys_id	int(5)	Ei	K	
unixtime	timestamp	Ei		CURRENT_TIMESTAMP

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
vuodenXluvut	year(4)	Ei		
answer1	int(10)	Kyllä		NULL
answer2	int(10)	Kyllä		NULL
answer3	tinyint(1)	Kyllä		NULL
answer4	int(10)	Kyllä		NULL
answer5	int(10)	Kyllä		NULL
answer6	int(10)	Kyllä		NULL
answer7	int(10)	Kyllä		NULL
answer8	int(10)	Kyllä		NULL
answer9	int(10)	Kyllä		NULL
answer10	int(10)	Kyllä		NULL
answer11	int(10)	Kyllä		NULL
answer12	varchar(255)	Kyllä		NULL
answer13	varchar(255)	Kyllä		NULL
answer14	varchar(255)	Kyllä		NULL
answer15	varchar(255)	Kyllä		NULL
answer16	int(10)	Kyllä		NULL
answer17	int(10)	Kyllä		NULL
answer18	int(10)	Kyllä		NULL
answer19	int(10)	Kyllä		NULL
answer20	int(10)	Kyllä		NULL
answer21	int(10)	Kyllä		NULL
answer22	int(10)	Kyllä		NULL
answer23	int(10)	Kyllä		NULL
answer24	int(10)	Kyllä		NULL
answer25	int(10)	Kyllä		NULL
answer26	int(10)	Kyllä		NULL

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
answer27	int(10)	Kyllä		NULL
answer28	int(10)	Kyllä		NULL
answer29	int(10)	Kyllä		NULL
answer30	int(10)	Kyllä		NULL
answer31	int(10)	Kyllä		NULL
answer32	int(10)	Kyllä		NULL
answer33	int(10)	Kyllä		NULL
answer34	varchar(255)	Kyllä		NULL
answer34a	varchar(255)	Kyllä		NULL
answer35	int(10)	Kyllä		NULL
answer36	varchar(255)	Kyllä		NULL
answer37	int(10)	Kyllä		NULL
answer38	int(10)	Kyllä		NULL
answer39	int(10)	Kyllä		NULL
answer40	int(10)	Kyllä		NULL

Taulukossa 3 on esitetty questionoptions-tilukko, joka sisältää vastausvaihtoehdot lomakkeen valinta- tai monivalintakysymyksiin.

Taulukko 3. questionoptions-tilukko

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
id	int(5)	Ei	K	
QID	int(5)	Ei		
qop1	varchar(255)	Ei		
qop2	varchar(255)	Ei		
qop3	varchar(255)	Ei		

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
qop4	varchar(255)	Ei		
qop5	varchar(255)	Ei		
qop6	varchar(255)	Ei		
qop7	varchar(255)	Ei		
qop8	varchar(255)	Ei		

Taulukossa 4 on esitetty questions-tilukko, joka sisältää kysymyksen, kysymyksen vastaustyyppin, sijainnin lomakkeen jollain sivulla ja sivun jossain kohdassa.

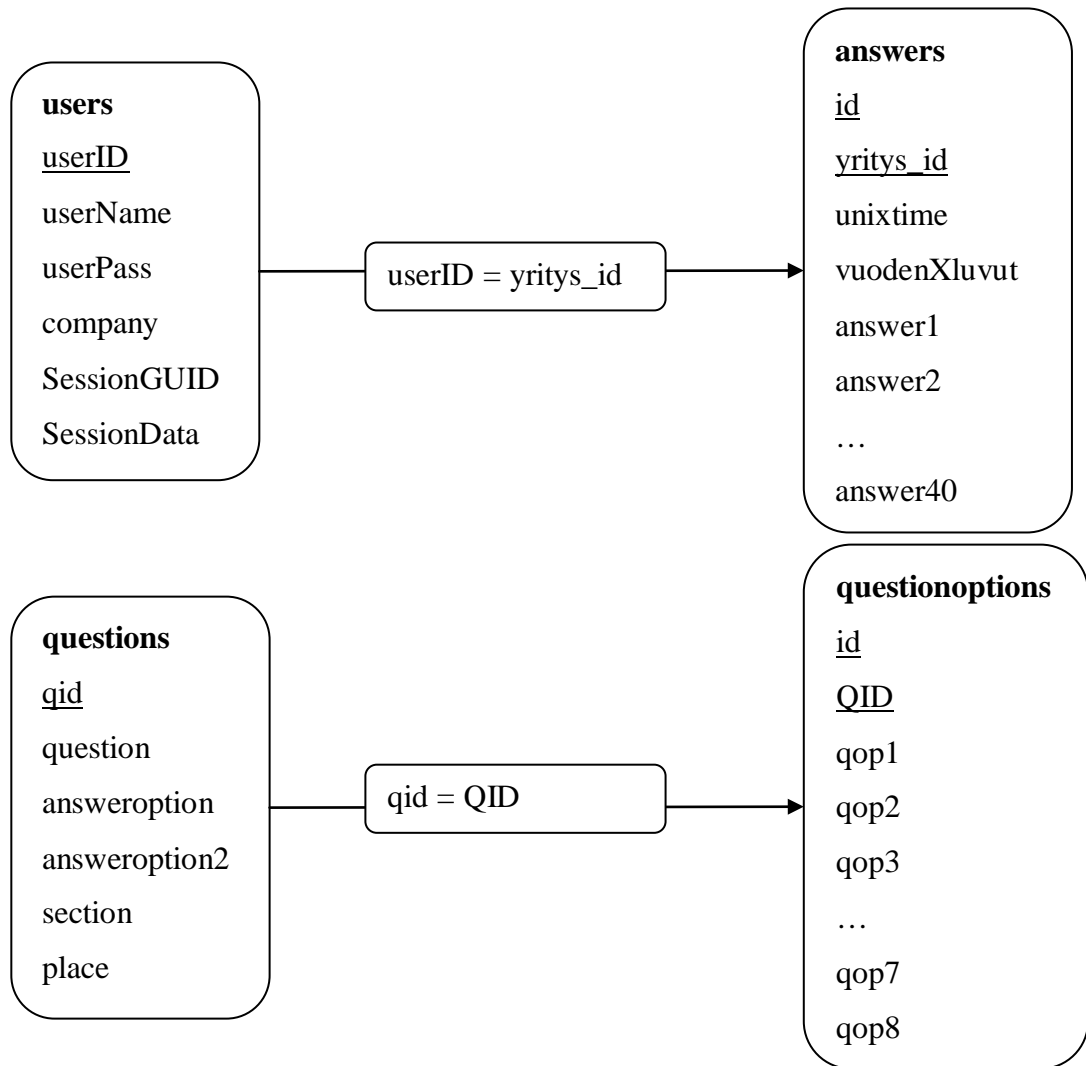
Taulukko 4. questions-tilukko

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
qid	int(5)	Ei	K	
question	varchar(255)	Ei		
answeroption	int(2)	Ei		
answeroption2	(1)	Ei		
section	int(2)	Ei		
place	int(2)	Ei		

Taulukossa 5 on esitetty users-tilukko, joka sisältää käyttäjän yksilöivän id-tunnuksen, käyttäjätunnuksen, suojatun salasanan, yrityksen nimen ja kirjautumissession tunnistet.

Taulukko 5. users-tilukko

Sarake	Tyyppi	Tyhjä	Avain	Oletusarvo
userID	int(10)	Ei	K	
userName	varchar(255)	Ei		
userPass	varchar(255)	Ei		
company	varchar(255)	Ei		
SessionGUID	varchar(32)	Kyllä		<i>NULL</i>
SessionData	mediumtext	Kyllä		<i>NULL</i>



Kuvio 22. Tietokannan rakenne

Kuviossa 22 käydään läpi miten tietokantataulukot ovat yhteydessä toisiinsa. Ylemmässä osassa nähdään linkitys users- ja answers-tilukon välillä. Taulukossa users voi olla vain yksi käyttäjä, mutta yhdellä käyttäjällä voi olla useita vastauksia. Alemmassa osassa nähdään linkitys questions- ja questionoptions-tilukoiden välillä. Kysymyksiä voi olla useita, mutta yhdessä kysymyksessä on olemassa vain yhdet vastausvaihtoehdot.

Jotta MySQL -tietokantaa voidaan käyttää Internet-sivun pohjana, täytyy tehdä linkki PHP:lla näiden kahden välille (kuvio 23).

```
<?php

define('SQL_USER', 'käyttäjätunnus');
define('SQL_PASS', 'salasana');
define('SQL_DB', 'tietokanta');

// Linkki tietokantapalvelimelle
$link = mysql_connect('localhost', SQL_USER, SQL_PASS);
if(!$link) :
    die('Ei voi yhdistää: ' . mysql_error());
endif;

// Valitaan tietokanta missä taulukot ovat
$db = mysql_select_db(SQL_DB, $link);
if(!$db) :
    die ('Ei voi yhdistää tietokantaan: ' . mysql_error());
endif;
?>
```

Kuvio 23. PHP-kielen ja MySQL-tietokannan yhteyden luonti

Jos nämä toimivat oikein, saadaan luotua yhteys tietokantapalvelimelle ja haluttuun tietokantaan. Tämän jälkeen kaikki kyselyt tietokannasta voidaan suorittaa pelkällä PHP-koodilla esimerkiksi: `mysql_query(SELECT max(answer2) FROM answers)`.

4.4.2 Etusivu

Etusivu on luotu käyttäen taulukkoa sivupohjana. Sivun on jaettu neljään osaan: otsikko, tietojensyöttö, omat tiedot, grafiikat. Otsikko sisältää luonnollisesti sivun otsikon, johon vaihtuu käyttäjän nimi kirjautumisen mukaan. Tietojensyöttöosio sisältää aktiiviset linkit, jos joltain vuodelta on jäänyt tiedot antamatta, muuten linkit on poistettu käytöstä.

Omat tiedot -osa sisältää linkit joka vuodelle jolloin tiedot on annettu, joista näkee omat syötetyt tiedot. Grafiikat osio sisältää linkit jo syötettyjen tietojen graafisiin esityksiin.

4.4.3 Syöttölomake

Syöttölomake sisältää samantyyppisen lomakkeen kuin liitteessä 1. Lomake on vain jaettu osiin prosesseittain. Lomake esitetään myöhemmin kappaleessa 5.

4.4.4 Grafiikkasivu

Tämä sivu sisältää Flash-sovelluksen, jolla graafiset kaaviot voidaan esittää. Aikaisemmin käsittelinkin jo miten tehdään Flash-kaavioita. Yksinkertaisesti kerrottuna halutut tiedot haetaan MySQL-tietokannasta PHP:n avulla. Kun nämä tiedot saadaan muokattua JSON-formaattiin, voidaan nämä liittää osaksi sivun lähdekoodia ja saada näin ollen Flash-kaavio.

Osa tuloksista on kuitenkin tekstimuodossa joten nämä pitää näyttää erillisellä ”tyhjä” sivulla. Grafiikkasivu esitetään myöhemmin kappaleessa 5.

5 TULOKSET

5.1 Järjestelmän käyttö

Järjestelmän koeversio löytyy osoitteesta <http://asennusrumba.com/Satakunta> käyttäjä-tunnuksilla testi ja salasanalla testi.

Järjestelmään on oikeus vain sallituilla henkilöillä/tahoilla, joten järjestelmä aukeaa ensin kirjautumissivulle (kuvio 24).

Käyttäjän kirjautuminen

Käyttäjä:

Salasana:

Kuvio 24. Järjestelmään kirjautuminen

Järjestelmän tietokannassa on joka käyttäjälle oma käyttäjä ja salasana yhdistelmä, joista salasana on suojattu md5-salauksella. Kun tiedot on syötetty ja painetaan Login, järjestelmä tarkastaa löytyykö käyttäjälle vastaavaa salasanan hash-tunnistetta. Jos vastaavaa tunnistetta ei löydy, palataan takaisin kirjautumissivulle. Jos vastaava tunnistetta löytyy, päästetään etusivulle. Järjestelmä on turvallinen, koska jokainen sivu tarkistaa käyttäjän kirjautumistiedot, jos niitä ei löydy, palataan kirjautumissivulle.

5.2 Järjestelmän etusivu

Etusivu koostuu kolmesta alueesta: Tietojen syöttö, tietojen katselu, kaaviot. Syötä tiedot -alueesta pääsee syöttämään kyseisen vuoden tiedot järjestelmään. Jos tiedot on jo syötetty, linkki näkyy harmaana, kuten kuviosta 25 voidaan nähdä. Linkit ovat kytkettynä tietokantaan. Jos tietokannasta löytyy kyseisen vuoden tiedot, linkki tulostetaan

harmaana eli deaktivoituna linkkinä. Omat tiedot -alue sisältää linkit niiden vuosien tietoihin, jotka on järjestelmään annettu. Nämä linkit lisääntyvät sen mukaan mikä vuosi tiedot on annettu. Grafiikat -alue sisältää linkit menneiden vuosien kaavioihin.

Kirjaudu ulos	Tervetuloa Toriseva Oy Satakunnan elintarvikeohjelman seurantajärjestelmään	
Syötä tiedot	Katso Omat tiedot	Katso grafiikat
2007	Vuoden 2007 tiedot	Vuoden 2007 grafiikat
2008	Vuoden 2008 tiedot	Vuoden 2008 grafiikat
2009	Vuoden 2009 tiedot	Vuoden 2009 grafiikat

Kuvio 25. Järjestelmän etusivu

5.3 Järjestelmän syöttölomake

Tietojen syöttölomake on kuin normaali lomake. Yrityksen nimi ja päiväys tulevat automaattisesti kirjautumistiedoista. Lomakkeet sisältävät avoimia kysymyksiä ja valintakysymyksiä. Lomakkeiden yksiköt ovat yleensä kappaleita tai euroja kysymyksestä riippuen. Lomakkeet tallentavat tiedot tietokantaan joka sivun jälkeen. Viimeisen sivun linkistä kiitetään vastauksista ja palataan etusivulle. Syöttölomake on esitetty kuviossa 26-31.

Yritys:	Toriseva Oy
Aika:	07.08.2010
Henkilöstön käyttö (henkilötyövuotta)	
Liikevaihto €	
Päätoimiala	<input type="radio"/> Kala <input type="radio"/> Leipomotuotteet <input type="radio"/> Punainen liha <input type="radio"/> Vihannesten jalostus <input type="radio"/> Valkoinen liha (broiler/kalkkuna) <input type="radio"/> Viljellyt sienet
	Sivu 2/5

Kuvio 26. Yleistietojen syöttölomake sivu 1

Asiakasprosessi

Laskutusasiakkaiden lukumäärä	
Tehtyjen asiakastytyväisyyystutkimusten lukumäärä seurantavuonna	
Tehtyjen markkinatutkimusten lukumäärä seurantavuonna	
Asiakasreklamaatioiden määrä seurantavuonna (jättäkää tyhjäksi jos ette halua ilmoittaa)	
	Sivu 3/5

Kuvio 27. Asiakasprosessi syöttölomake sivu 2

Investoinnit tuotantoprosesseihin ja koneisiin	
Investoinnit infrastruktuuriin (rakennukset ym.)	
Informaatioteknologiaan (laitteet) käytetty rahamäärä	
Investoinnit ympäristötekniologiaan	
Sähköinen toiminnanohjausjärjestelmä: Kattavuus	<input type="checkbox"/> tuotanto <input type="checkbox"/> osto <input type="checkbox"/> myynti <input type="checkbox"/> taloushallinto <input type="checkbox"/> henkilöstöhallinto <input type="checkbox"/> kaikki
Sähköinen toiminnanohjausjärjestelmä: Käyttöönottovuosi	<input type="radio"/> ennen 1995 <input type="radio"/> 1995-2000 <input type="radio"/> 2001-2003 <input type="radio"/> 2004-2006 <input type="radio"/> 2007
Sähköinen toiminnanohjausjärjestelmä: Järjestelmän nimi	
Sähköisessä tilausjärjestelmässä mukanaolo	<input type="radio"/> yrityksemme on mukana jossakin sähköisessä tilausjärjestelmässä <input type="radio"/> ei ole vielä
	Sivu 4/5

Kuvio 28. Teknologiaprosessi syöttölomake sivu 3

Investoinnit tuotekehitykseen (reseptien kehitys, laboratoriokustannukset, arvioinnit, käytetty työaika jne.) seurantavuonna	
Tuotevalikoimassa olevien tuotteiden lukumäärä seurantavuonna	
Seurantavuonna tuotevalikoimaan tulleet uutuustuotteet	
Satakunnassa meneillään oleviin kehittämishankkeisiin mukanaolosta seurantavuonna aiheutuneet kustannukset yrityksellenne	
Markkinointipanostukset ja investoinnit tiedon hankintaan (mainonta, messumatkat, benchmarking, pilotititutukset jne) seurantavuonna	
Koulutuspäivien lukumäärä yrityksessänne seurantavuonna	
Koulutuksessa olleiden henkilöiden lukumäärä seurantavuonna	
Oppisopimuskoulutuksessa olevien määrä seurantavuonna (kpl)	
Muuhun kuin informaatioteknologian koulutukseen käytetty rahamäärä seurantavuonna	
Informaatioteknologiakoulutukseen käytetty rahamäärä seurantavuonna	
Informaatioteknologiaan (ohjelmistot) seurantavuonna käytetty rahamäärä	
Henkilöstön koulutustaso:	
Henkilöstön koulutustaso: korkeakoulu/ yliopistotutkinto	
Henkilöstön koulutustaso: ammattikorkeakoulututkinto	
Henkilöstön koulutustaso: opistotason tutkinto	
Henkilöstön koulutustaso: ammatillinen tutkinto	
Henkilöstön koulutustaso: ei tutkintoa	
Henkilöstön koulutustaso: opiskelija	
Työkyvyn (TYKY) ylläpitämiseen seurantavuonna käytetty rahamäärä	
Koulutuksen pääaiheet seurantavuonna	<input type="checkbox"/> valmistusprosessi/tuotanto/teknologia <input type="checkbox"/> markkinointi ja myynti <input type="checkbox"/> taloushallinto <input type="checkbox"/> työssäjaksaminen <input type="checkbox"/> huolto ja kunnossapito <input type="checkbox"/> työsuojelu ja ergonomia <input type="checkbox"/> informaatioteknologia Muu
	Sivu 5/5

Kuvio 29. Liiketoimintaprosessi syöttölomake sivu 4

Talousprosessi

Viennin määrä €	
Markkina-alueet	<input type="checkbox"/> Paikalliset toimitukset <input type="checkbox"/> Koko Satakunta <input type="checkbox"/> Useampi maakunta <input type="checkbox"/> Koko Suomi <input type="checkbox"/> Pohjoismaat <input type="checkbox"/> Eurooppa <input type="checkbox"/> muu alue
Käyttökate-%	
Sijoitetun pääoman tuotto-%	
Koko pääoman tuotto-%	
Omavaraisuusaste-%	
Tallenna vastaukset	

Kuvio 30. Järjestelmän syöttölomake sivu 5

**Kiitoksia vastauksistanne.
Siirrämme teidät kohta etusivulle.**

Kuvio 31. Vahvistus ja kiitos

5.4 Järjestelmän Omat tiedot-sivu

Kun etusivulla valitaan Omat tiedot, avautuu valitulta vuodelta syötetyt tiedot. Ylärivistä nähdään linkit edellisvuoden ja seuraavan vuoden tietoihin sekä päästään suoraan kyseisen vuoden kaavioihin (kuvio 32).

Etusivulle	Edellisen vuoden tiedot	Seuraavan vuoden tiedot	Kaaviot tämän vuoden tiedoista
Yrityksen seurantavuoden 2007 tiedot:			
Henkilöstön käyttö (henkilötyövuotta)			
27			
Liikevaihto €			
100000			
Päätoimiala			
Punainen liha			
Laskutusasiakkaiden lukumäärä			
15			
Tehtyjen asiakastytytyväisyyystutkimusten lukumäärä seurantavuonna			
25			
Tehtyjen markkinatutkimusten lukumäärä seurantavuonna			
5			
Asiakasreklamaatioiden määrä seurantavuonna (jättäkää tyhjäksi jos ette halua ilmoittaa)			
12			
Investoinnit tuotantoprosesseihin ja koneisiin			
12000			
Investoinnit infrastruktuuriin (rakenukset ym.)			
13000			
Informaatioteknologiaan (laitteet) käytetty rahamäärä			
14000			
Investoinnit ympäristöteknologiaan			
15000			
Sähköinen toiminnanohjausjärjestelmä: Kattavuus			
osto,taloushallinto			
Sähköinen toiminnanohjausjärjestelmä: Käyttöönottovuosi			
2001-2003			

Kuvio 32. Järjestelmän Omat tiedot-sivu

5.5 Järjestelmän kaaviosivusto

Kun etusivulla valitaan grafiikat, aukeaa kaaviosivu, josta valitaan mikä aiemmin kysytystä kysymyksistä halutaan esittää graafisena (kuvio 33).

Etusivulle	2007	2008	2009	2010	2011	2012	2013
Valitse kysymys:							
Valitse vertailtava kysymys ▼							

Kuvio 33. Järjestelmän Kaaviot-sivu

[Etusivulle](#) [2007](#)[2008](#)[2009](#)[2010](#)[2011](#)[2012](#)[2013](#)

Valitse kysymys:

Valitse vertailtava kysymys

TAUSTATIEDOT

Henkilöstön käyttö (henkilötyövuotta)

Liikevaihto €

Päätoimiala

ASIAKASPROSESSI

Laskutusasiakkaiden lukumäärä

Tehtyjen asiakastytytyväisyyssytutkimusten lukumäärä seurantavuonna

Tehtyjen markkinatutkimusten lukumäärä seurantavuonna

Asiakasreklamaatioiden määrä seurantavuonna (jättäkää tyhjäksi jos ette halua ilmoittaa)

TEKNOLOGIAPROSESSI

Investoinnit tuotantoprosesseihin ja koneisiin

Investoinnit infrastruktuuriin (rakennukset ym.)

Informaatioteknologiaan (laitteet) käytetty rahamäärä

Investoinnit ympäristötekнологiaan

Sähköinen toiminnanohjausjärjestelmä: Kattavuus

Sähköinen toiminnanohjausjärjestelmä: Käyttöönottovuosi

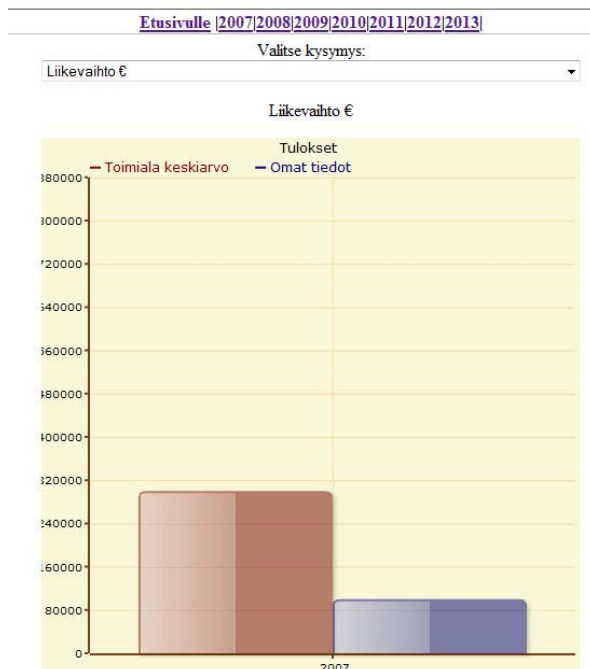
Sähköinen toiminnanohjausjärjestelmä: Järjestelmän nimi

Sähköisessä tilausjärjestelmässä mukanaolo

LIIKETOIMINTAPROSESSI

Kuvio 34. Kaaviot-sivun kysymykset/vastaukset

Valitaan kysymys (kuvio 34), jonka vastaukset halutaan nähdä. Järjestelmä hakee tiedot tietokannasta ja etsii näistä tiedoista suurimman arvon, yrityksen toimialan, toimialan keskiarvon ja askelluksen kaaviota varten. Askellukseen käytetään kymmenesosa suurimmasta arvosta. Lopputulos graafisena kuviona on esitetty kuviossa 35.



Kuvio 35. Järjestelmän kaavio liikevaihdosta

Kaikista numeraalista dataa sisältävistä kysymyksistä voidaan esittää graafinen kaavio. Ei-numeraalinen data esitetään normaalina tekstinä.

6 JOHTOPÄÄTÖKSET JA YHTEENVETO

Opinnäytetyön ongelmana oli suunnitella ja toteuttaa Satakunnan elintarvikeohjelman seurantajärjestelmä, jolla seurataan eri toimialayritysten kuluja vuosittain. Järjestelmä luotiin Internet-ympäristöön.

Opinnäytetyön empiirisessä osiossa kerroin Satakunnan elintarvikejärjestelmän toteutuksesta prosessina. Työssä käytiin läpi koko toteutusprosessi aina määrittelystä, suunnittelun kautta järjestelmän toteutukseen ja valmiiseen tuotteeseen.

Työn teoriaosassa tarkastelin järjestelmän suunnitteluun liittyviä asioita, esittelin PHP/MySQL-ympäristöä sekä Internet-ympäristöä ja tutustuttiin Flash-grafiikan käyttöön.

Opinnäytetyön tuloksena saatiin luotua järjestelmä Internet-ympäristöön, jolla opinnäytetyölle asetettu tavoite saavutettiin. Aina ohjelmointityössä on tietenkin jotakin parannettavaa. Työhön valittu PHP- ja MySQL-tuki ympäristölle toimi loistavasti.

Ainoan ongelman tuotti kaaviosovelluksen lisäys Internet-ympäristöön. Vaikkakin ohjeet sovelluksen käyttöön olivat yksinkertaiset, mutta tietojen saaminen oikeassa muodossa tietokannasta sovellukselle oli useiden pulmien takana. Esimerkiksi tietokannasta haettava arvojoukko ei sopinut suoraan kaaviosovelluksen käyttöön.

Kunnollisen Flash-tuen puuttuminen kaaviosovelluksesta työtä tehdessä jätti osan kaavioiden teksteistä vertikaaliseen asentoon tilan puutteen vuoksi.

7 TULEVAISUUDEN NÄKYMÄ

Koska graafiset kaaviot saadaan nyt suoraan tietokoneelta, niin aiemmin samaan työhön käytetyt henkilöt voidaan sijoittaa muihin töihin. Ja näin ollen yrityksen tulos paranee, koska ei ole turhaa työvoimaa analysoimassa yritysten tietolomakkeita.

8 LOPPUTULOKSEN ARVIOINTI

Työn tavoitteet tulivat suoritettua. Järjestelmä toimii, kaaviot toimivat. Aikaa työn tekemiseen meni odotettua enemmän, johtuen osin täyspäiväisestä työstä tämän opinnäytetyön ohessa. Järjestelmän kaavio-ominaisuutta tehdessä ongelmana oli Flash-sovelluksen yhteensovittaminen MySQL-tietokannan kanssa, johon vierähti välillä päivä, ellei jopa viikkokin. Järjestelmä on, muutamista takaiskuista huolimatta, erittäin hyvä kokonaisuus yrityksille jotka tekevät asiakaskyselyitä vuosittain tai jopa useamman kerran vuodessa.

Jos aloittaisin alusta, tutkisin Open Flash Chart-sovelluksen toimintaperiaatteet ja muokkaisin sovelluksen toimimaan haluamallani tavalla. Seuraava opiskelija, joka alkaisi tämän tyyppistä työtä tehdä, voisi suunnitella jo paperille valmiiksi, minkälainen järjestelmän tulisi lopuksi olla. Tämä auttaa paljon kaavioiden ohjelmoinnissa ja dynaamisten sivujen toiminnassa.

LÄHDELUETTELO

Asleson, R. & Schutta, N.T. 2007. Ajax. Readme.fi.

FusionCharts Technologies LLP. 2009. FusionCharts v3. [verkkodokumentti]. [viitattu 5.10.2010] Saatavissa: <http://www.fusioncharts.com>

Glazebrook, J. 2009. Open Flash Chart. [verkkodokumentti]. [päivitetty 15.5.2009]. [viitattu 22.5.2009]. Saatavissa: <http://teethgrinder.co.uk/open-flash-chart/>

Glazebrook, J. 2009. Open Flash Chart 2 tutorial. [verkkodokumentti]. [viitattu 5.10.2010]. Saatavissa: <http://teethgrinder.co.uk/open-flash-chart-2/tutorial.php>

Immonen, J. 2003. Johdatus ohjelmistotuotantoon. [verkkodokumentti]. [päivitetty 18.3.2003]. [viitattu 5.10.2010]. Saatavissa: http://cs.joensuu.fi/~jimmonen/jot_moniste/jot_moniste_121.html

JSON. 2010. JSON. [verkkodokumentti]. [viitattu 8.8.2010]. Saatavissa: <http://www.json.org/>

Kolehmainen, K. 2006. PHP & MySQL Teoriasta käytäntöön. Readme.fi.

Lindberg, H. 2003. Extreme Programming. [verkkodokumentti]. [päivitetty 8.8.2010]. [viitattu 9.8.2010] Saatavissa: http://www.cs.uta.fi/research/thesis/masters/Lindberg_Harri.pdf

Peltomäki, J. 2005. JavaScript. Docendo.

Peltomäki, J. & Nykänen, O. 2006. Web-selainohjelmointi. Docendo.

Pham, Michael. 2010. phpDesigner 7. [verkkodokumentti]. [viitattu: 5.10.2010]. Saatavissa: <http://www.mpsoftware.dk/phpdesigner.php>

PHP Group. 2009. PHP -ohjelmointikielen kirjasto. [verkkodokumentti]. [päivitetty 15.5.2009]. [viitattu 22.5.2009]. Saatavissa: <http://www.php.net>

Rantala, Ari. 2005. Web-ohjelmointi. Docendo.

Wikipedia. 2010a. PHP. [verkkodokumentti]. [viitattu 8.8.2010]. San Francisco: Wikimedia Foundation Inc. Saatavissa: <http://fi.wikipedia.org/wiki/PHP>

Wikipedia. 2010b. MySQL. [verkkodokumentti]. [viitattu 8.8.2010]. San Francisco: Wikimedia Foundation Inc. Saatavissa: <http://fi.wikipedia.org/wiki/Mysql>

Kaikki luvut koskevat seuranta- vuotta 2006:		
	Pekka Kuisma	
Vastausaika	28.08.2008	
Henkilöstön käyttö (henkilötyövuotta):	Henkilötyövuodet:	<input type="text"/> hlötyöv
	<input type="text"/>	<input type="text"/>
Liikevaihto €	<input type="text"/> €	
Päätoimiala	<input type="checkbox"/> Kala <input type="checkbox"/> Leipomotuotteet <input type="checkbox"/> Punainen liha <input type="checkbox"/> Vihannesten jalostus <input type="checkbox"/> Valkoinen liha (broiler/kalkkuna) <input type="checkbox"/> Viljellyt sienet	
ASIAKASPROSESSI		
Laskutusasiakkaiden lukumäärä	<input type="text"/>	kpl
Tehtyjen asiakastytyväisyystutkimusten lukumäärä seurantavuonna	<input type="text"/> 0	kpl
Tehtyjen markkinatutkimusten lukumäärä seurantavuonna	<input type="text"/> 0	kpl
Asiakasreklamaatioiden määrä seurantavuonna	<input type="text"/>	kpl <input type="checkbox"/> ei tietoa
TEKNOLOGIAPROSESSI		
Investoinnit tuotantoprosesseihin ja koneisiin	<input type="text"/>	€
Investoinnit infrastruktuuriin (rakennukset ym.)	<input type="text"/>	€
Informaatioteknologiaan (laitteet) käytetty rahamäärä	<input type="text"/>	€
Investoinnit ympäristöteknoologiaan:	<input type="text"/>	€
Sähköinen toiminnanohjausjärjestelmä:	Kattavuus: <input type="checkbox"/> tuotanto <input type="checkbox"/> osto <input type="checkbox"/> myynti	

	<input type="checkbox"/> taloushallinto <input type="checkbox"/> henkilöstöhallinto <input type="checkbox"/> kaikki osa-alueet
	Käyttöönottovuosi: <input type="checkbox"/> ennen 1995 <input type="checkbox"/> 1995-2000 <input type="checkbox"/> 2001-2003 <input type="checkbox"/> 2004-2006 <input type="checkbox"/> 2007
	Järjestelmän nimi: <input type="text"/>
	<input type="text"/>
Sähköisessä tilausjärjestelmässä mukanaolo:	<input type="checkbox"/> yrityksemme on mukana jossakin sähköisessä tilausjärjestelmässä <input type="checkbox"/> ei ole vielä
LIIKETOIMINTAPROSESSI	
Investoinnit tuotekehitykseen (reseptien kehitys, laboratoriokustannukset, arvioinnit, käytetty työaika jne.) seurantavuonna:	<input type="text"/> €
Tuotevalikoimassa olevien tuotteiden lukumäärä seurantavuonna:	<input type="text"/> kpl
Seurantavuonna tuotevalikoimaan tulleet uutuustuotteet:	<input type="text"/> kpl
Satakunnassa meneillään oleviin kehittämiss-hankkeisiin mukanaolosta seurantavuonna aiheutuneet kustannukset yrityksellenne	<input type="text"/> €
Markkinointipanostukset ja investoinnit tiedon hankintaan (mainonta, messumatkat, benchmarking, pilottitutustumiset jne) seurantavuonna:	<input type="text"/> €
Koulutuspäivien lukumäärä yrityksessänne seurantavuonna	<input type="text"/> kpl
Koulutuksessa olleiden henkilöiden lukumäärä seurantavuonna:	<input type="text"/> kpl
Oppisopimuskoulutuksessa olevien määrä seurantavuonna (kpl):	<input type="text"/> kpl
Muuhun kuin informaatioteknologian koulutukseen käytetty rahamäärä seurantavuonna:	<input type="text"/> €
Informaatioteknologiakoulutukseen käytetty rahamäärä seurantavuonna:	<input type="text"/> €
Informaatioteknologiaan (ohjelmistot) seurantavuonna käytetty rahamäärä	<input type="text"/> €

Henkilöstön koulutustaso:	korkeakoulu/ yliopistotutkinto	<input type="text"/> kpl
	ammattikorkea- koulututkinto	<input type="text"/> kpl
	opistotason tutkin- to	<input type="text"/> kpl
	amatillinen tutkinto	<input type="text"/> kpl
	ei tutkintoa	<input type="text"/> kpl
	opiskelija	<input type="text"/> kpl
Työkyvyn (TYKY) ylläpitämiseen seuranta- vuonna käytetty rahamäärä:	<input type="text"/> €	
Koulutuksen pääaihealueet seurantavuonna:	<input type="checkbox"/> valmistusprosessi/tuotanto/teknologia <input type="checkbox"/> markkinointi ja myynti <input type="checkbox"/> taloushallinto <input type="checkbox"/> työssäjaksaminen <input type="checkbox"/> huolto ja kunnossapito <input type="checkbox"/> työsuojelu ja ergonomia <input type="checkbox"/> informaatioteknologia	
	Muita aiheita, mitä? <div> <input type="text"/> </div>	
TALOUSPROSESSI		
Viennin määrä €	<input type="text"/> €	
Markkina-alueet:	<input type="checkbox"/> Paikalliset toimitukset <input type="checkbox"/> Koko Satakunta <input type="checkbox"/> Useampi maakunta <input type="checkbox"/> Koko Suomi <input type="checkbox"/> Pohjoismaat <input type="checkbox"/> Eurooppa <input type="checkbox"/> muu alue	

Käyttökate-%	<div><div></div></div> %
Sijoitetun pääoman tuotto-%	<div><div></div></div> %
Koko pääoman tuotto-%	<div><div></div></div> %
Omavaraisuusaste-%:	<div><div></div></div> %